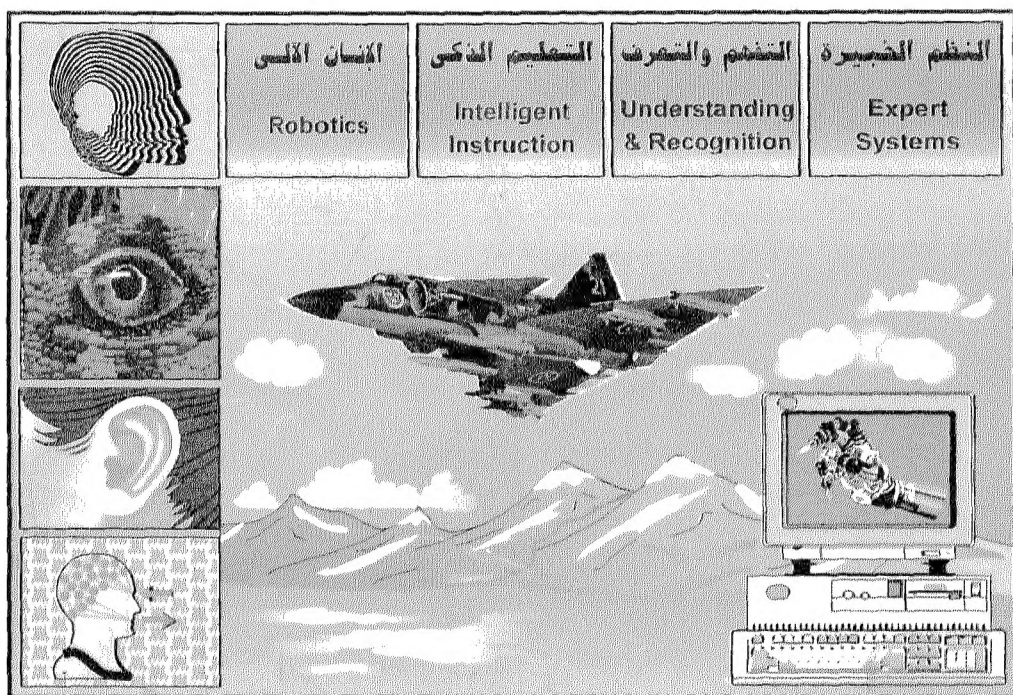


الذكاء الاصطناعي والشبكات العصبية

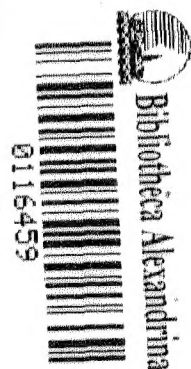


الأستاذ الدكتور محمد على الشرقاوي

الكتاب الأول

ضمن سلسلة

علوم وتكنولوجيا حاسبات المستقبل



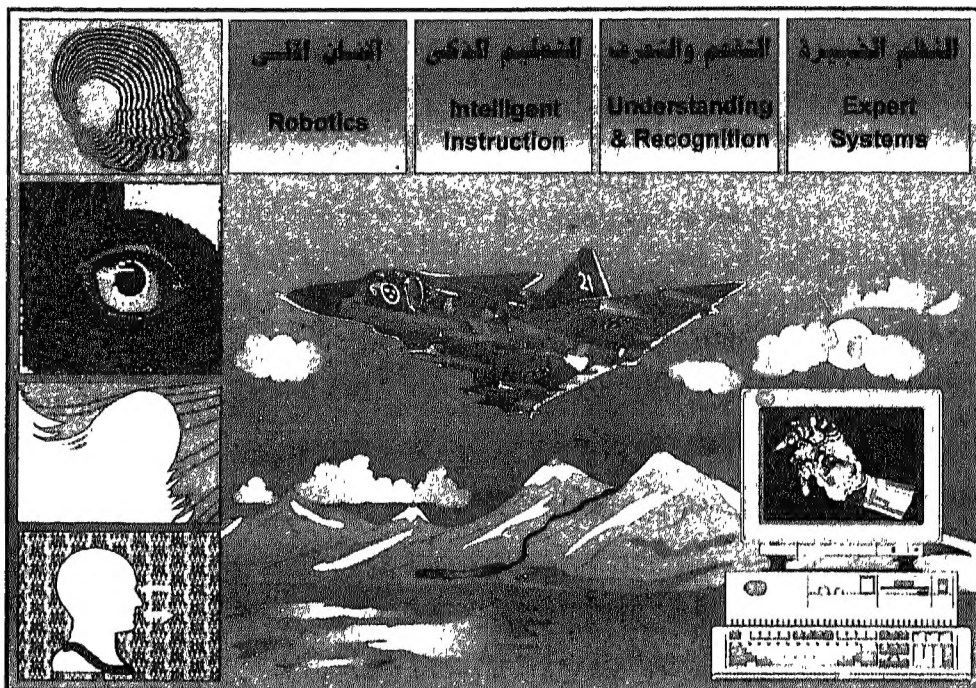
Artificial Intelligence Computer Center



مركز الذكاء الاصطناعي للحاسبات

الذكاء الاصطناعي
والشبكات العصبية

الذكاء الاصطناعي والشبكات العصبية



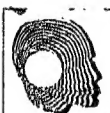
الأستاذ الدكتور محمد علي الشرقاوي

الكتاب الأول

ضمن سلسلة

علوم وتكنولوجيا حاسبات المستقبل

Artificial Intelligence Computer Center



مركز الذكاء الاصطناعي للحاسبات

© حقوق النشر

All Rights Reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the Author.

لا يجوز نشر أى جزء من هذا الكتاب أو اختزان مادته بطريقة الاسترجاع، أو نقله على أى وجه ، أو بأى طريقة ، سواء كانت إلكترونية ، أو ميكانيكية ، أو بالتصوير ، أو بالتسجيل ، أو خلاف ذلك إلا بموافقة المؤلف على هذا كتابة و مقدما".

رقم الإيداع ٣٠٨٤ / ٩٦

تقديم ونظرة مستقبلية

Introduction and Futuristic Look

يعرف الانسان الذكى بأنه ذلك الانسان القادر على وضع الحلول للمشكلات بشكل منطقى صائب وسريع، كما يعرف الانسان الخبير بأنه الانسان القادر على استخدام خبرته المكتسبة فى مجال معين لايجاد الحل السليم لمشكلة من المشاكل الصعبة والتي لا يمكن حلها بدون الخبرة التى اكتسبها فى هذا المجال المعين، وبذلك يتميز الانسان بكثير من الصفات المعتمدة على الذكاء الانسانى والتي ترتبط بقدرته على التفاعل والاستجابة مع البيئة المحيطة به واستخدام الاستدلال لايجاد ووضع الحلول للمشاكل التى تقابله فى الحياة اليومية والتي ترتبط بالدرجة الاولى بالتحدث باللغات الطبيعية والتفهم لما يسمعه الانسان من احاديث والنظر الى الاشياء والتعرف عليها والتمييز بين الاشكال المتشابهة والقدرة على الاسترجاع والتذكر والتفكير والترجمة من لغة الى اخرى واستخدام الخبرة المكتسبة فى اتخاذ القرار والاحساس وغير ذلك من الصفات الاخرى العديدة، ومن هنا تم اطلاق صفات كثيرة على الانسان المتميز فى حل المشكلات مثل الانسان الذكى او الماهر او الخبير او ذو الدهاء فى حل الامور او ذو الفراسة وغير ذلك من الصفات الاخرى.

من المعروف كذلك أن الانسان يعتمد أساسا فى المعالجة على التمثيل الشكلى او الرمزي للاشياء وصفاتها والعلاقات بينها والتي تختلف كثيرا عن المعالجة الرقمية التى تعتمد عليها معظم نظم الحاسبات الشائعة الاستعمال هذه الايام والتي تتخذ معمارية فون نيومان اساسا لها والقادرة والمتفوقة فى حل المشكلات الحسابية والعلمية والتجارية والتطبيقات الأخرى التى تتطلب سرعة الحساب والفرز والترتيب والحزن والاسترجاع مثل قواعد البيانات مثلا، وبذلك نرى ان الانسان يتفوق على الحاسب فى المعالجة الرمزية او الشكلية وحل المشكلات التى تتطلب ذكاءا أو خبرة مكتسبة. ومن هنا نشأت الحاجة الى نقل نظم وسائل ذكاء وتفوق الانسان مثل المعالجة الرمزية وطرق الاستدلال وغير ذلك من السلوك الانسانى الى نظم الحاسبات لكى تبنى قدرا من الذكاء عند معالجتها لمشكلات البيئة المحيطة مثل التحدث والتفهم باللغات الطبيعية وغير ذلك من المجالات التى يتفوق فيها الانسان الذكى، وبذلك تبلورت نظما وتقنيات منقولة من الانسان الى الحاسب اطلق عليها الذكاء الاصطناعى.

ويشتمل هذا الكتاب على شرح لأساسيات الذكاء الاصطناعى والتي تعبر عن الجيل الخامس لتكنولوجيا الحاسبات والتي تختلف فى الخواص والمضمون عن الأجيال السابقة التى اعتمدت أصلا على نوعية المكونات المادية الرقمية للنظم الحاسوبية حيث يستخدم الذكاء الاصطناعى المعالجة الرمزية والتي يتميز بها الجيل الخامس علاوة على الإمكانيات المتاحة للنظم الرقمية والتي تفى بالتطبيقات التجارية والمحاسبية، ويحتوي هذا الكتاب على أربعة أجزاء، يتناول الجزء الاول التعريف بالذكاء الاصطناعى وعلاقته بالذكاء الانسانى واساسياته ومجالاته المختلفة والنظم الخبيرة ومجالاتها المختلفة وبعض النظم التطبيقية وقائمة النظم الخبيرة الشائعة، ونظم الحاسبات المعتمدة على تقنيات الذكاء الاصطناعى وحاسبات الجيل الخامس للمشروع اليابانى.

ويتناول الجزء الثانى اسس المعالجة الرمزية والنمذجة الحاسوبية وقشيل المعارف وآليات وتقنيات الاستدلال مثل المنطق الاسنادى ومحددات الكمية والاثبات التحليلى والتوحيد ونظم الاستدلال المعتمدة على القواعد مثل نظم الانتاج ومحرك الاستدلال ونظم التسلسل المختلفة. ويتناول الجزء الثالث لغات البرمجة والتي تشمل المعالجة الرمزية والمعالجة المختلطة واساسيات لغة البرمجة المنطقية السريعة ولغة البرمجة باسلوب القائمة وبرمجة العلاقات والصفات واستخدام تطبيقات من الحياة اليومية تتطلب استخدام الذكاء فى تنفيذها والمعالجة الرمزية للعمليات الحاسوبية والرسوم التصويرية وقواعد البيانات الديناميكية، وعرض بعض النظم التطبيقية.

يتناول الجزء الرابع اساسيات الشبكات العصبية والحساب العصبى وكيف تطورت والسمات العامة للنيرون والنماذج الحاسوبية المختلفة وميكانيكية التعلم والشبكات العصبية المتعددة الطبقات وسلوكها واستخداماتها فى بعض التطبيقات مثل نظم النشر الالكترونى لقواعد البيانات والمخطط التسويقي لشركات الطيران. كما تم عرض أساسيات شبكات كوهن ذاتية التعليم والتنظيم الذاتى وطرق التعليم باستخدام التوزيع الكمى للمتجهات وبعض تطبيقاتها مثل النشر المكتبى باستخدام الصوت كما تناول اساسيات شبكات هوبفيلد ذات الاتصال الكامل وميكانيكية الحزن والتذكر وآلة بولتز مان الحرارية وبعض التطبيقات، وفى نهاية هذا الجزء تم عرض لنظرية الرنين المتكيف التى تعتبر اساسا للشبكات المكثفة التوازي وعرض النماذج المختلفة لها، كما يتميز الكتاب بعرض الترجمة الانجليزية للمصطلحات بجانب النص العربى لها مما يؤدى الى سهولة الفهم ويساعد على ربط المادة العلمية بالاتجاه العالمى فى هذا التخصص، كما يتولى عرض التطبيقات

المرتبطة بالحياة اليومية والبيئية لكى يصل القارئ الى اهمية استخدام تقنيات الذكاء الاصطناعى والشبكات العصبية الاصطناعية.

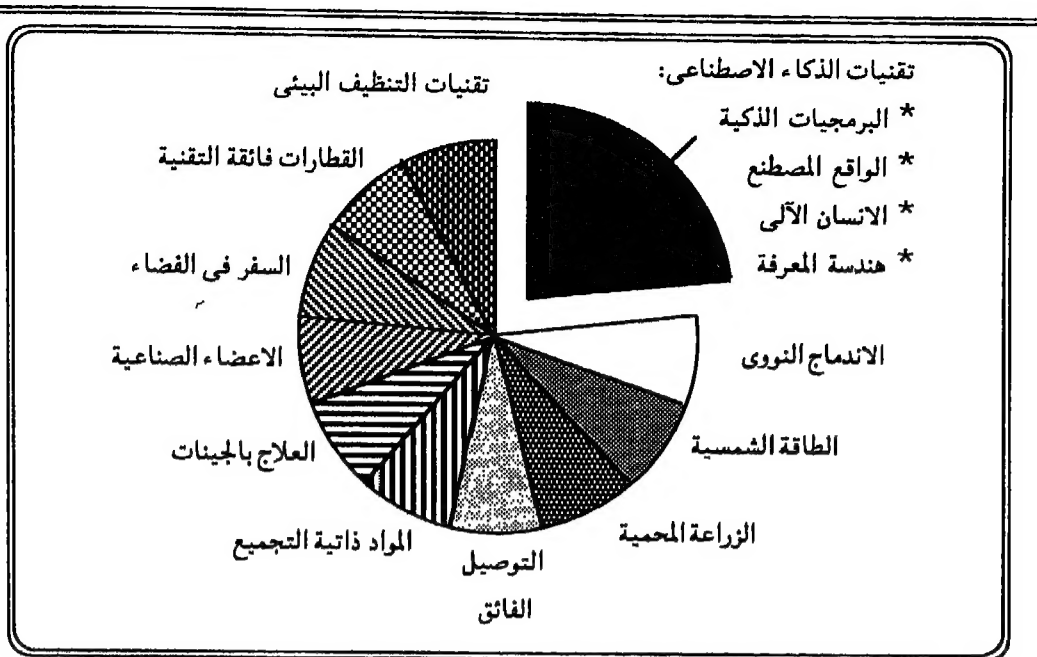
نظرة مستقبلية

لقد أثبتت الأبحاث العلمية أن جيل الذكاء الاصطناعى سوف يصل إلى ما بعد المعالجة الرقمية والرمزية ومن المتوقع أن تتميز الفترة من نهاية التسعينات من هذا القرن وبداية القرن القادم (١٩٩٥ - ٢٠٠٥ م) بالتقدم في المحاور الآتية :

يمثل المحور الاول المعالجة الرمزية (Symbolic Processing) حيث يتم إستخدام الرموز بكفاءة فى المعالجة والوصول إلى الحل الكامل لمشكلة ما والذي يتمثل نجاحه فى التمثيل الرمزي الدقيق لسلوك الذكاء الإنسانى عند محاولة حل هذه المشكلة ووضعها بشكل مناسب فى برامج ونظم الحاسبات علاوة على أن تكون المعالجة الرقمية جزء من هذه البرامج وذلك بفضل إستخدام المعالجة المختلطة.

ويمثل المحور الثانى المعالجة المتوازية (Parallel Processing) والتي تشتمل على استخدام المعالجة الرمزية والمعالجة الرقمية والإستعانة بتكنولوجيا المعالجة المتوازية لتحليل موضوعات فى الزمن الحقيقى والتي تفى بأغراض التحليل المباشر (On -line Analysis) .

ويمثل المحور الثالث الحساب العصبى (Neural Computing) حيث إن التطور الهائل والسريع للحساب العصبى بإستخدام أساليب تكنولوجيا المعالجة بالشبكات العصبية الاصطناعية (Artificial Neural Networks) سوف يؤدي إلى ظهور وتطور نظاماً للذكاء الاصطناعى والنظم الحبييرة تعتمد على التلاحم بين أساليب المعالجة الرمزية للذكاء الاصطناعى مع المعالجة بإستخدام الحساب العصبى، ويعتبر العقد الاخير من هذا القرن (١٩٠٠-٢٠٠٠م) هو عقد تطور الشبكات العصبية الاصطناعية والذي دعا بعض علماء الحاسبات إلى القول بأنها سوف تمثل الجيل السادس للحاسبات بالرغم من اعتمادها على المعالجة الرقمية وطرق الإستدلال التكاملى المتوازي، ويقدر الخبراء أن تقنيات الذكاء الاصطناعى سوف تمثل حوالى ٢٥٪ من مجمل التقنيات التى سوف تسيطر على مقاليد الامور العالمية فى القرن القادم كما هو مبين فى الشكل الموضوع وتشمل هذه التقنيات تقنيات البرمجيات الذكية وهندسة المعرفة والانسان الآلى



التقنيات المسيطرة فى القرن القادم

(الروبوتات) والنظم الخبيرة والواقع المصطنع والتي سوف تغطى الكثير من نواحي الحياة المختلفة، وسوف يكون للواقع المصطنع أهمية كبرى فى التفاعل مع الانسان حيث يقوم الحاسب بإدخال حواس الانسان مثل النظر والسمع وحاسة اللمس داخل البرنامج عن طريق وصلات مواءمة مناسبة لكي يعمل المستخدم وكأنه جزء من البرنامج، وبذلك يصبح الخيال المصطنع واقعا ملموسا.

وفيما يلى بعض التطورات المنتظرة فى ثلاث فروع من ميادين اساسية للتطبيقات الحيوية فى هذا المجال حيث تمثل اللغات الطبيعية الميدان الاول ويشمل التقدم فى هذا المضمار إستحداث نظم لتفهم الحديث المتصل فى بعض المجالات الخاصة، والترجمة الفورية للحديث من خلال الشبكات التليفونية لبعض اللغات، والتعرف الاوتوماتيكي على النصوص وإدخالها فى قواعد البيانات. ثم تتطور نظم التفهم للحديث المتصل والترجمة الفورية للحديث والتعرف الاوتوماتيكي للنصوص لتشمل جميع المجالات التى سوف يتم تدعيمها من خلال انشاء قواعد البيانات التى تعمل باللغات الطبيعية والانتاج الرخيص للمكونات الالكترونية للذاكرة وتقديم تقنيات تمثيل المعارف وطرق الاستدلال المختلفة وطرق التعلم وذلك للوصول الى نظم حاسبات

تكون لغتها أحسن بكثير من لغة بعض الأفراد حيث تكون قادرة على الحوار والتفهم والتحدث والترجمة وما الى ذلك.

وتمثل النظم الخبيرة الميدان الثانى حيث يتم التوسع فى توظيف النظم الخبيرة كأدوات لاتخاذ القرار فى ميادين وموضوعات محدودة مع زيادة الاعتماد على مداخل المواءمة التى تعمل باللغات الطبيعية، وفى المراحل الاولى يتم التكامل بين النظم الخبيرة فى شبكات متكاملة فمثلا تتكون الشبكة الطبية من عدة نظم خبيرة يتخصص كل منها فى مجال طبي معين، وتصبح الشبكة بيت خبرة عالية المستوى وتترابط الشبكات لتشمل جميع الميادين، ويجرى تطوير ما يعرف بالآلات الذكية والتى تصل الى مستوى ذكاء يتم توليفة من خلال الاتصال المباشر بين الالة والانسان ويذهب بعض العلماء الى القول بان تبادل المعرفة يمكن ان يتم بالاتصال المباشر بين المخ الانسانى والنظام الخبير وذلك بزراعة شرائح الكترونية داخل الانسان تكون قادرة علي نقل المعرفة الفورية وعند ذلك تصبح الالة الذكية قادرة على الأداء المماثل للانسان.

ويمثل علم الروبوتات (الأنسنة) الميدان الثالث بتطور الانسان الآلى حيث تتمكن بعض الروبوتات من التحرك ويجرى الاعتماد عليها بشكل مكثف فى وسائل الانتاج المختلفة ولا ينتظر ان تحل مشكلة التحرك والرؤية كاملا وتظل دون المستوى الانسانى فى اوائل القرن القادم وسوف تعمل الروبوتات فى الحياة العامة مثل التحصيل وقيادة المركبات وفى المنازل، ثم تتطور لتصبح حرية التحرك كاملة للروبوتات التى تكون كاملة التجهيز والتى تحمل على متنها الحاسب وكذلك وسائل الرؤية الالكترونية المتكاملة والتى تكون دقتها فى كثير من الاحوال أكثر بكثير من دقة الانسان.

والله ولى التوفيق

المؤلف

أ.د. محمد على الشرقاوى

الصفحة	الموضوع	مسلسل
محتويات الكتاب		
الجزء الأول		
الذكاء الاصطناعي الحاضر والمستقبل		
الفصل الأول " الذكاء الاصطناعي والذكاء الإنساني		
Artificial and Human Intelligence		
٢١		(١-١) مقدمة
٢٢		(٢-١) العقل الإنساني ومعالجة المعرفة
٢٤		(٣-١) المخ والتمثيل الرمزي للمعرفة
٢٥		(٤-١) مراحل تطور الذكاء الاصطناعي
٢٦		(١-٤-١) المرحلة الاولى
٢٦		(٢-٤-١) المرحلة الثانية
٢٧		(٣-٤-١) المرحلة الثالثة (١٩٧٥-١٩٩٥)
٢٧		(٤-٤-١) المرحلة المستقبلية (١٩٩٥-٢٠٢٢م)
٢٩		(٥-١) العلاقة بين الذكاء البشري والاصطناعي
٣٢		(٦-١) أساسيات نظم الذكاء الاصطناعي
٣٥		
الفصل الثاني " مجالات الذكاء الاصطناعي "		
Domains of Artificial Intelligence		
٣٩		(١-٢) مجالات الذكاء الاصطناعي
٤٠		(٢-٢) النظم الخبيرة (Expert Systems)
٤١		(٣-٢) إثبات النظريات آلياً (Automated Theorem Proving)
٤١		(٤-٢) التفهم والتعرف على اللغات الطبيعية (Natural Language Understanding)
٤٣		(١-٤-٢) نظام تفهم للحديث او الكلام (Speech Understanding System)
٤٤		(٢-٤-٢) الترجمة الآلية للحديث والنص.
٤٦		(٣-٤-٢) الأسلوب التتابعي العام للتفهم والتعرف
٤٦		(٤-٤-٢) إنتاج البرمجيات آلياً (Automated Software Generation)
٤٩		(٥-٢) علم الروبوتات (الأنسنة) (Robotics)
٤٩		(٦-٢) تمثيل المعارف آلياً (Automated Knowledge Representation)
٥٢		(٧-٢) التعليم والتعلم باستخدام الحاسبات (Computer - Assisted Learning)
٥٣		

الصفحة	الموضوع	مسلسل
٥٤	(Intelligent Computer Aided Instruction) التعليم الذكى باستخدام الحاسبات	(١-٧-٢)
٥٨	(Design Issues for (I C A I) Systems) مقومات تصميم نظم التعليم الذكى	(٢-٧-٢)
٥٩	(Multimedia) الوسائط المتعددة	(٨-٢)
٦٣	الفصل الثالث " النظم الخبيرة ومجالاتها المختلفة " Expert Systems and its Different Domains	
٦٤	(Expert Systems) النظم الخبيرة	(١-٣)
٦٦	(١-١-٣) محاكاة للنظام الخبير " المستشار الطبى "	
٦٨	(٢-١-٣) محاكاة لنظام خبير مبسط للتعرف على الصور الجوية	
٦٩	(٣-١-٣) محاكاة لنظام خبير لتحديد الأعطال (Fault Diagnostic Expert System)	
٧٢	(٢-٣) الخصائص والمتطلبات العامة للنظم الخبيرة	
٧٢	(١-٢-٣) السمات العامة	
٧٣	(٢-٢-٣) المجالات المناسبة	
٧٣	(٣-٢-٣) تجمع النتائج من البيانات (Data Convergence)	
٧٥	(٣-٣) تقنيات البرمجة للنظم الخبيرة (Expert System Techniques)	
٧٧	(٤-٣) اللغات والحزم المناسبة للنظم الخبيرة	
٨٠	(٥-٣) أدوات بناء النظم الخبيرة (Expert System Building Tools)	
٨٣	(٦-٣) الأدوات المساعدة (ToolKits)	
٨٥	الفصل الرابع " المجالات التطبيقية للنظم الخبيرة " Expert System's Practical Applications	
٨٦	(١-٤) المجالات التطبيقية للنظم الخبيرة	
٨٨	(٢-٤) قائمة النظم الخبيرة (Expert Systems List)	
٩٥	الفصل الخامس " حاسبات الجيل الخامس " Fifth Generation Computer Systems	
٩٦	(١-٥) نظم الحاسبات (Computer Systems)	
٩٧	(٢-٥) نظم الحاسبات المعتمدة على الذكاء الاصطناعى (AI- Based Computer Systems)	
٩٨	(١-٢-٥) مقومات حاسبات الذكاء الاصطناعى (Issues for AI- Based Computer)	
١٠٠	(٣-٥) حاسبات الجيل الخامس المشروع اليابانى (Fifth Generation Computer Systems)	

الصفحة	الموضوع	مسلسل
١٠٠	الخطة الأساسية للمشروع	(١-٣-٥)
١٠٢	نتائج المشروع	(٤-٥)
١٠٢	مشروعات الجيل الخامس الياباني (١٩٩٢-١٩٨١)	(١-٤-٥)
١٠٢	حسابات الإستدلال المتوازية	(٢-٤-٥)
الجزء الثاني		
المعالجة الرمزية		
Symbolic Processing		
١٠٧	الفصل السادس " الأسس الرياضية للمعالجة الرمزية"	
Mathematical Basics For Symbolic Processing		
١٠٨	الأسس الرياضية للمعالجة الرمزية	(١-٦)
١٠٨	المعرفة - الوصف والعلاقات.	(١-١-٦)
١٠٩	تصنيف العلاقات	(٢-١-٦)
١٠٩	العلاقات والصفات	(٣-١-٦)
١١٠	العلاقات غير الثنائية (ذات الرتبة أكبر من ٢).	(٤-١-٦)
١١١	الشبكات الدلالية (Semantic Nets)	(٢-٦)
١١٣	الفصل السابع "النمذجة الحسابية وتمثيل المعارف"	
Computational Modelling and Knowledge Representation.		
١١٤	النمذجة الحسابية (Computational Modelling)	(١-٧)
١١٥	مراحل إنشاء وتطوير النماذج (Levels of Model Development)	(١-١-٧)
١١٧	خواص التمثيل لنظم الإستدلال المبني على النماذج (MBR Characteristics)	(٢-٧)
١١٨	القواعد الهندسية للنمذجة	(٣-٧)
١١٩	التمثيل باستخدام الإطارات (Representation using frames)	(٤-٧)
١٢٠	تصميم الإطارات	(١-٤-٧)
١٢١	التعرف على طرق تمثيل المعرفة (Knowledge Representation Approaches)	(٥-٧)
١٢٣	الفصل الثامن "آليات تقنيات الإستدلال (المنطق الرمزي الحسابي)"	
Toolbox of Inference Techniques (Symbolic Computational Logic)		
١٢٤	المنطق الرمزي الحسابي (Symbolic Computational Logic)	(١-٨)
١٢٥	الجملة (Statement)	(١-١-٨)

الصفحة	الموضوع	مسلسل
١٢٦	(٢-٨) التعبير الرمزي عن الجملة	
١٢٧	(٣-٨) قاعدة التضمن الشرطي (Implication)	
١٢٨	(١-٣-٨) التضمن والنظم المبنية على القواعد (Implication and Rule - Based Systems)	
١٢٨	(٢-٣-٨) خواص قاعدة التضمن الشرطي	
١٢٩	(٤-٨) التسلسل المتقدم (Forward Chaining)	
١٣٠	(٥-٨) شبكات الاستدلال (Inference Nets)	
١٣١	(٦-٨) قاعدة التضمن الشرطي الإيجابي (Modus Ponens)	
١٣١	(٧-٨) الإسنادات و المتغيرات (Predicates and Variables)	
١٣١	(٨-٨) المنطق الإسنادي (Predicate Logic)	
١٣٢	(١-٨-٨) الإسنادات المعتمدة على الأرقام (Numerically Based Predicates)	
١٣٢	(٢-٨-٨) الإسناد ذو المتغير الواحد (Unary Predicate)	
١٣٣	(٣-٨-٨) اختيار التمثيل للإسنادات (Choosing Predicate Representation)	
١٣٣	(٤-٨-٨) الإسنادات و الجمل المركبة (Predicates and Compound Statement)	
١٣٤	(٩-٨) محددات الكمية (Quantifiers)	
١٣٥	(١-٩-٨) تكافؤ الإسنادات المحددة الكمية (Equivalence of Quantified Predicates)	
١٣٥	(٢-٩-٨) التكرار باستخدام النفي (Tautology)	
١٣٦	(٣-٩-٨) اللغة المنطقية أحادية الرتبة	
١٣٧	(١٠-٨) إيجاد الحلول للمشكلات (Problem Solving)	
١٣٧	(١-١٠-٨) الإثبات التحليلي أو التفنيد (Proof by Resolution (Refutation))	
١٤٠	(١١-٨) التوحيد (Unification)	
١٤٠	(١-١١-٨) قواعد التوحيد (Unification Rules)	
١٤١	(٢-١١-٨) التعويض و الربط (Substitution and Binding)	
١٤١	(٣-١١-٨) خوارزميات التوحيد (Unification Algorithms)	
١٤٦	(٤-١١-٨) التوحيد بمقارنة قوائم الربط (Unification by Checking the Binding Lists)	
١٤٩	الفصل التاسع" نظم الاستدلال المعتمدة على القواعد"	
	Rule-Based Inference Systems	
١٥٠	(١-٩) نظم الإنتاج (Production Systems)	
١٥١	(١-١-٩) محرك الاستدلال أو الاستنباط (Inference Engine)	
١٥٢	(٢-١-٩) السمات العامة لنظم الانتاج (Features of Pruduction Systems)	

الصفحة	الموضوع	مسلسل
١٥٢	(Inference Engine & Problem Solving) محرك الإستدلال وإيجاد الحل للمشكلات	(٣-١-٩)
١٥٣	(Production System Properties) خصائص نظم الانتاج	(٤-١-٩)
١٥٤	(Production System and Planning) نظم الانتاج والتخطيط	(٥-١-٩)
١٥٦	(Decomposable Production System) نظم الانتاج القابلة للتجزئة	(٦-١-٩)
١٥٦	(Rule-Based Chaining Inference) القواعد المعتمدة على القواعد	(٢-٩)
١٥٧	(Forward Chaining System using LISP) نظام تسلسل امامى باستخدام اللىسب	(١-٢-٩)
١٥٨	(Backward Chaining) التسلسل الخلفى	(٣-٩)
١٥٨	(Basis for Backward Chaining) أساسيات التسلسل الخلفى	(١-٣-٩)
الجزء الثالث		
لغات البرمجة والتطبيقات		
Programming Languages and Applications		
١٦٣	الفصل العاشر " الذكاء الاصطناعى ولغات البرمجة "	—
(AI and Programming Languages)		
١٦٤	(Symbolic Processing & Programming Languages) المعالجة الرمزية ولغات البرمجة	(١-١٠)
١٦٤	(Symbolic and " Conventional " Processing) المعالجة الرمزية المعالجة التقليدية	(١-١٠)
١٦٤	(Coupled Processing Systems) نظم المعالجة المختلطة	(٢-١٠)
١٦٦	(Manipulable Representations & Data Structures) التمثيل الحسابى وتراكيب البيانات	(٣-١٠)
١٦٧	(Descriptive Concept for Prolog) الفرض الوصفى للبرمجة المنطقية	(٤-١٠)
١٦٨	(٥-١٠) الشروط الواجبة في لغات الذكاء الاصطناعى	
١٦٩	(Environmental Requirements for AI Programming) المتطلبات البيئية لنظم البرمجة	(٦-١٠)
١٧٠	(Programming Languages Classification) تصنيف لغات البرمجة	(٧-١٠)
١٧٥	الفصل الحادى عشر "تطبيقات باستخدام اللغات المنطقية (البرولوج السريع)"	
Application using Prolog (Turbo Prolog)		
١٧٦	(Prolog) لغة البرمجة المنطقية	(١-١١)
١٧٦	(Turbo Prolog) لغة البرمجة المنطقية السريعة	(٢-١١)
١٧٧	(٣-١١) التمثيل الرمزي للكيانات والعلاقات	
١٧٧	(١-٣-١١) التمثيل الرمزي للنشاط الرياضى	

الصفحة	الموضوع	مسلسل
١٧٩	(Variables) المتغيرات (٢-٣-١١)	
١٧٩	(Objects and Relations) الأشياء والعلاقات (٣-٣-١١)	
١٨٠	(Compound Goals) الاهداف المركبة (٤-١١)	
١٨٠	قاعدة بيانات للسيارات المستخدمة (١-٤-١١)	
١٨١	(Anonymous Variable) المتغير المجهول (٢-٤-١١)	
١٨٢	(Backtracking) التتبع الخلفي (٥-١١)	
١٨٢	(١-٥-١١) التمثيل الرمزي للصفات الشخصية	
١٨٣	(Tautology Using Not) (لا) النفي باستخدام الاداة (٦-١١)	
١٨٣	(Matchmaker) (١-٦-١١) التوافق بين العادات للأشخاص	
١٨٤	(٢-٦-١١) تمثيل وبرمجة العلاقات العائلية	
١٨٧	(٣-٦-١١) تمثيل وبرمجة الهوايات الشخصية	
١٨٧	(٧-١١) المجالات القياسية	
١٨٨	(١-٧-١١) استخدام المجالات القياسية	
١٨٩	(Compound Objects) المركبة (٨-١١) تمثيل توصيف الأشياء	
١٩١	(١-٨-١١) المجالات المشتملة على تفصيلات دقيقة للكائنات المركبة	
١٩٣	(Recursion) التكرار (٩-١١)	
١٩٤	(Recursive Objects) المتكررة (١-٩-١١) الأشياء	
١٩٥	(Lists) القوائم (١٠-١١)	
١٩٨	(Search Mechanism) البحث (١١-١١) ميكانيكية	
١٩٨	(Unification Process) التوحيد (١-١١-١١) عملية	
١٩٨	(٢-١١-١١) قاعدة بيانات للمطبوعات بالمكتبات	
٢٠٠	(Controlling the Search for Solutions) (١٢-١١) التحكم في البحث لاجراز الاهداف	
٢٠٣	(Forced Backtracking) (١٣-١١) التتبع الخلفي المدفوع	
٢٠٤	(Preventing Backtracking) (١-١٣-١١) الايقاف المحدد للتتبع الخلفي	
٢٠٧	الفصل الثاني عشر" المعالجة الرمزية للعمليات الحسابية والرسوم التصويرية" Symbolic Processing for Arithmetic Operations & Graphics	
٢٠٨	(Symbolic Processing for Arithmetic Operations) (١-١٢) المعالجة الرمزية للعمليات الحسابية	
٢١٠	(٢-١٢) إيجاد الحل لمعادلة الدرجة الثانية	
٢١٢	(Arithmetic Functions) (٣-١٢) الدوال الحسابية	

الصفحة	الموضوع	مسلسل
٢١٣	الفصل الثالث عشر "قواعد البيانات الديناميكية" Dynamic Databases	
٢١٤	(١-١٣) قواعد البيانات الديناميكية (Dynamic Databases)	
٢١٤	(١-١-١٣) توصيف قاعدة البيانات	
٢١٥	(٢-١-١٣) التعامل مع الحقائق (Handling Facts)	
٢١٦	(٢-١٣) قواعد البيانات الديناميكية الممتدة	
٢١٨	(٣-١٣) المميزات العامة للغة البرمجة المنطقية السريعة	
٢١٩	(١-٣-١٣) عناصر لغة البرمجة المنطقية السريعة	
٢٢١	الفصل الرابع عشر "بناء نظم تطبيقية" Building Application Systems	
٢٢٢	(١-١٤) نظام خيرة تعليمي مبسط للتعرف على الحيوانات (Small Expert System for Animal Identification)	
٢٢٦	(٢-١٤) نظام أولي لتقدير افضل المسارات بين المدن (Prototyping for Best Routing between Cities)	
٢٢٧	(٣-١٤) (نظم الانقاذ) اختيار المسار الآمن (Safe Routing Selection System)	
٢٣٠	(٤-١٤) محاكاة الدوائر المنطقية الرقمية (Simulation of Digital Logic Circuits)	
٢٣١	(٥-١٤) حل الالغاز (Puzzle Solving)	
٢٣٣	(٦-١٤) تخمين الكلمات (Word Guessing)	
٢٣٥	الفصل الخامس عشر "لغة البرمجة بأسلوب القائمة" LISP	
٢٣٦	(١-١٥) التعرف على لغة البرمجة بالقائمة (List Processing)	
٢٣٧	(١-١-١٥) الذرات (Atoms) والقوائم (Lists)	
٢٣٧	(٢-١٥) تراكيب البرامج (Programming Structure)	
٢٣٨	(٣-١٥) العمليات الحسابية المنطقية (Arithmetic Manipulation)	
٢٣٩	(١-٣-١٥) التعامل بالقوائم (List Operations)	
٢٤١	(٤-١٥) الدوال الأكثر استخداماً (Commonly Used LISP functions)	
٢٤٦	(٥-١٥) التكرار عند عكس عناصر القائمة (Recursive List reverser)	
٢٤٧	(٦-١٥) التصنيف والتوحيد للنماذج (Classification /Model Unification)	
٢٤٨	(٧-١٥) خواص التمثيل بلغة الليسب (Characterization of LISP Representations)	
٢٤٩	(١-٧-١٥) نماذج ذاكرة الحاسب (Computer Memory Models)	

مستسل	الموضوع	الصفحة
٢٥٠	(٢-٧-١٥) تمثيل الرسوم البيانية بالذاكرة (Computer Representation of Graphs)	
٢٥١	(٣-٧-١٥) تمثيل العلاقات بالمصفوفات (Matrix Representation of Relations)	
٢٥٣	(٨-١٥) خلية أداة الربط (cons) وبناء القوائم (The cons Cell and Building Lists)	
٢٥٥	(٩-١٥) التعبير النقطي أو الثنائيات المنقطه (LISP "Dot " or Dotted Pair Notation)	
الجزء الرابع		
الشبكات العصبية الاصطناعية والحساب العصبي		
Artificial Neural Network & Neural Computing		
٢٥٧	الفصل السادس عشر "تطور الشبكات العصبية الاصطناعية"	
Artificial Neural Networks Development		
٢٥٨	(١-١٦) الشبكات العصبية الاصطناعية (Artificial Neural Networks)	
٢٦١	(٢-١٦) تطور الشبكات العصبية الاصطناعية (Artificial Neural Networks Development)	
٢٧٠	(٣-١٦) السمات العامة للنيرون (General Features of Neuron)	
٢٧٠	(١-٣-١٦) التركيب المبسط للوحدة العصبية النيرون (Simple Structure of Neuron)	
٢٧٢	(٢-٣-١٦) ميكانيكية التعلم في النظم البيولوجية	
	(Learning Mechanism in Biological Syst.)	
٢٧٣	(٤-٣-١٦) نموذج "ماك كلوش وبس" الرياضي (McCulloch & Pitts Mathematical Model)	
٢٧٤	(٥-٣-١٦) النموذج ذو القيمة المنحازة (Biased Model)	
٢٧٦	(٤-١٦) ميكانيكية التعلم في النيرونات البسيطة (Learning Mechanism in Simple neurons)	
٢٧٨	(١-٤-١٦) خوارزم التعلم لعنصر الادراك (Perceptron Learning Algorithm)	
٢٨١	(٢-٤-١٦) التمثيل المتجهي لميكانيكية التعلم	
	(Vectorial Analogy for Learning Mechanism)	
٢٨٢	(٣-٤-١٦) حدود التطبيق لنموذج عنصر الادراك المفرد	
	(Limitations of Perceptron Applications)	
٢٨٣	الفصل السابع عشر "الشبكات العصبية المتعددة الطبقات"	
Multilayer Neural Networks		
٢٨٤	(١-١٧) الشبكات العصبية المتعددة الطبقات (Multilayer Neural Network)	
٢٨٥	(١-١-١٧) النموذج متعدد الطبقات (Multilayer Model)	
٢٨٦	(٢-١-١٧) قاعدة دلتا المعممة للتعليم (Generalised Learning Delta Rule)	
٢٨٧	(٣-١-١٧) التمثيل الرياضي لقاعدة دلتا للتعليم (Mathematical Repres of Delta Rule)	
٢٩٢	(٤-١-١٧) خوارزم التعليم للنموذج متعدد الطبقات (The Multilayer Learning Algorithm)	

الصفحة	الموضوع	متسلسل
٢٩٣	(Solution Of (XOR) Problem) وضع الحل لمشكلة الدائرة (أو- المانة)	(٥-١-١٧)
٢٩٤	(Behaviour Of Multilayer Network) سلوك الشبكات المتعددة الطبقات	(٢-١٧)
٢٩٦	(Multilayer Network As Classifiers) الشبكات المتعددة الطبقات كمصنفات	(٣-١٧)
٢٩٧	(Convex Hulls and Convex Regions) الحدود والمناطق المحدبة	(١-٣-١٧)
٣٠٠	(Function Determination of Layers) تحديد طبيعة العمل للطبقات	(٢-٣-١٧)
٣٠٠	(Generalisation Property) خاصية التعميم	(٤-١٧)
٣٠١	(Fault Tolerance) التجاوز عن الخطأ	(٥-١٧)
٣٠١	(Some Applications) بعض تطبيقات الشبكات المتعددة الطبقات	(٦-١٧)
٣٠١	(NETTALK) الشبكة الناطقة	(١-٦-١٧)
٣٠٢	(Airline Marketing Tactican) المخطط التسويقي لشركات الطيران	(٢-٦-١٧)
٣٠٣	(ECG Noise Filtering) تنقية رسم القلب الكهربائي من التشويش	(٣-٦-١٧)
٣٠٤	(Electronic Data Publishing System) نظام النشر الإلكتروني للبيانات	(٤-٦-١٧)
٣٠٧	الفصل الثامن عشر "شبكات كوهن العصبية ذاتية التنظيم" Kohonen Self - Organising Neural Networks	
٣٠٨	(Unsupervised Learning) التعليم الذاتي	(١-١٨)
٣٠٨	(Self-organisation Concepts) مفاهيم التنظيم الذاتي	(١-١-١٨)
٣٠٩	(Two-dimensional Kohonen's Network) شبكة كوهن ذات البعدين	(٢-١-١٨)
٣١٠	(Kohonen's Algorithm for Self-organised Learning) خوارزم كوهن للتعليم الذاتي	(٢-١٨)
٣١٢	(Lateral Interconnection) التوصيل التبادلي الجانبي	(١-٢-١٨)
٣١٥	(Learning by Vector Quantisation (LVQ)) التعليم باستخدام التوزيع الكمي للمتجهات	(٣-١٨)
٣١٥	(Some Applications) بعض تطبيقات شبكات كوهن	(٤-١٨)
٣١٥	(Phonetic Word- processing) النشر المكتبي باستخدام الصوت	(١-٤-١٨)
٣١٩	الفصل التاسع عشر "شبكات هوبفيلد العصبية ذات الإتصال الكامل" Hopfield Fully - Connected Neural Networks	
٣٢٠	(Hopfield Networks) شبكات هوبفيلد	(١-١٩)
٣٢٠	(Network Structure) تركيب (معمارية) الشبكة	(١-١-١٩)
٣٢١	(Behaviour Of Fully Connected Network) سلوك الشبكة ذات الإتصال الكامل	(٢-١٩)
٣٢٢	(Hopfield's Algori. for Autoassociative Network) خوارزم هوبفيلد للشبكات ذاتية الترابط	(٣-١٩)
٣٢٤	(Energy Landscape for Hopfield's Network) مستويات الطاقة لشبكات هوبفيلد	(٤-١٩)

الصفحة	الموضوع	مسلسل
٣٢٥	(Patterns Storing Mechanism) ميكانيكية الحزن للبصمات (١-٤-١٩)	
٣٢٧	(Stored Patterns Recall Mechanism) ميكانيكية استدعاء البصمات (التذكر) (٢-٤-١٩)	
٣٢٩	(Synchronous & Asynchronous Updating) التحديث المتزامن والغير متزامن (٣-٤-١٩)	
٣٣٠	(The Boltzmann Thermal Machine) آلة بولتزمان الحرارية (٥-١٩)	
٣٣٣	(The Boltzmann Learning Algori) خوارزم التعليم لآلة بولتزمان (١-٥-١٩)	
٣٣٥	(Some Application for Boltzmann Thermal Machine) بعض التطبيقات لآلة بولتزمان (٦-١٩)	
٣٣٥	(The Travelling Salesman Problem) مشكلة البائع المتجول (١-٦-١٩)	
٣٣٩	الفصل العشرون " نظرية الرنين المتكيف الشبكات العصبية المكثفة التوازي" Adaptive Resonance Theory (ART) Massively Parallel Neural Networks	
٣٤٠	(Adaptive Resonance Theory (ART)) نظرية الرنين المتكيف (١-٢٠)	
٣٤٠	(ART Network Architecture) معمارية شبكات نظرية الرنين المتكيف (٢-٢٠)	
٣٤٢	(Operation Mechanism For ART Nets) ميكانيكية العمل لشبكات نظرية الرنين المتكيف (٣-٢٠)	
٣٤٢	(Initialisation Phase) طور التنشيط (١-٣-٢٠)	
٣٤٣	(Recognition Phase) طور التعرف (٢-٣-٢٠)	
٣٤٥	(Comparison Phase) طور المقارنة (٣-٣-٢٠)	
٣٤٥	(Vigilance Threshold Test) حد مراقبة الاختبار (٤-٣-٢٠)	
٣٤٦	(Search Phase) طور البحث (٥-٣-٢٠)	
٣٤٦	(ART Network Learning Algorithm) خوارزم التعليم لنظرية الرنين المتكيف (٤-٢٠)	
٣٤٨	(Example of A Learning Cycle) مثال لدورة التعليم (١-٤-٢٠)	
٣٥١	قائمة المراجع	

الجزء الأول

الذكاء الاصطناعي الحاضر والمستقبل

Artificial Intelligence

The Present and The Future

الفصل الأول

الذكاء الاصطناعي

والذكاء الإنساني

Artificial and

Human Intelligence

(١-١) مقدمة (Introduction)

تشارك جميع الكائنات الحية فى وجود منظومات عصبية (Neural Systems) تمكنها من التعامل والتفاعل مع البيئة المحيطة بها كما تساعدنا فى التحكم فى العمليات الحيوية اللازمة لإستمرار الحياة لهذه الكائنات . وتختلف المنظومات العصبية من كائن إلى آخر حيث تكون بسيطة التركيب وطبيعة العمل فى الكائنات الأولية ذات التركيب الخلقى البسيط ومعقدة التركيب وطبيعة العمل فى الكائنات الأكثر علواً مثل الإنسان. وتعتبر المنظومة العصبية للإنسان أعقد المنظومات العصبية على الإطلاق والتي يتركز معظمها فى المخ البشرى الذى يتميز بطبيعة عمل أدت إلى تفوق الإنسان على سائر المخلوقات الأخرى فى قدرات التفهم والتعرف على الأشكال والرموز والتعلم والتحدث والتذكر والإدراك والسيطرة الدقيقة على الجهاز الحركى وما إلى ذلك من العديد من الصفات والقدرات التى لا يستطيع أى كائن آخر غير الإنسان إلى الوصول إليها.

كما يعرف الذكاء البشرى (Human Intelligence) بأنه المقدرة والمهارة على وضع وإيجاد الحلول للمشكلات (Problem Solving) بإستخدام الرموز (Symbols) وطرق البحث المختلفة و معالجة المعرفة (Knowledge) والقدرة على إستخدام الخبرة المكتسبة (Experties) فى اشتقاق معلومات ومعارف جديدة تؤدى إلى وضع الحلول لمشاكل ما فى مجال معين، ويتفاوت مستوى الذكاء من شخص إلى آخر كما يعتبر الذكاء البشرى هو المسئول عن التطور والإبداع فى فنى الحضارات المختلفة.

ونظراً لأهمية الذكاء البشرى فإن الإنسان كان ولا يزال دائم البحث عن طبيعة هذا الذكاء وكيف يمكن قياسه ووضع الخطوات لمحاكاة أساليبه فى شكل برامج بإستخدام الحاسبات . ولقد إقتصرت دراسة الذكاء البشرى لفترة طويلة على علماء النفس ، ولكن التقدم السريع فى جميع فروع العلوم فى النصف الأخير من هذا القرن قد أدى إلى مساهمة وتلاحم علوم كثيرة مثل الفسيولوجى والبيولوجى والرياضيات والفيزياء والهندسة والحاسبات والفلسفة واللغويات فى دراسة ومحاكاة نظم الذكاء الإنسانى وتطويرها ، فلقد راود الباحثين الأمل فى انتقال أساليب الذكاء الفطرى والخبرة المكتسبة للإنسان إلى نظم البرمجة للحاسبات لكى يمكن الإستفادة بهافى كثير من شتى مجالات الحياة المختلفة والتى تتطلب قدرا من الذكاء و الخبرة اللازمة لمسيرة

التطور فى التطبيقات الصناعية الزراعية والتجارية الحديثة . وبذلك أدى إستخدام الحاسبات فى مجالات التعرف على الأشكال والرموز والنماذج المختلفة إلى ظهور نظم الذكاء الاصطناعى والتي تميزت بانتقال جزء من اساليب الذكاء الإنسانى إلى نظم البرمجة للحاسبات والتي ساهمت بدورها فى بناء نظم الخبرة التى إشتملت بعضا من الخبرة المكتسبة للإنسان.

يقدم هذا الكتاب اساسيات الذكاء الاصطناعى والشبكات العصبية والنظم الخبيرة للقارىء العربى كما يحتوى على شرح للطاقت والقدرات للجيل الخامس للحاسبات والتي تعرف بحاسبات الاستدلال و يوضح أن الذكاء الاصطناعى هو بمثابة الثورة التى سببت الانفجار فى المعلومات والمعارف والتي واكبت التقدم فى التكنولوجيا الصناعية للحاسبات والذي يمكن ان يستهل بالاجابة على السؤال الآتى :

ما هو الذكاء الاصطناعى للحاسبات ؟

للإجابة عن هذا السؤال فإنه يمكن تعريف الذكاء الاصطناعى بأنه :

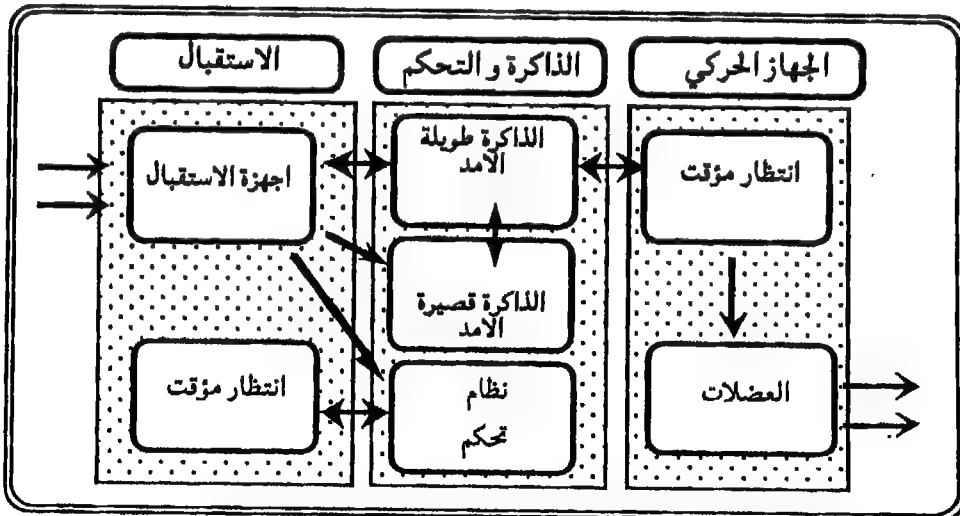
" ذلك الفرع من علوم الحاسب (Computer Science) الذى يمكن بواسطته خلق وتصميم برامج للحاسبات التى تحاكي اسلوب الذكاء الإنسانى لكى يتمكن الحاسب من أداء بعض المهام بدلا من الانسان والتي تتطلب التفكير والتفهم والسمع والتكلم والحركة " والتي ترجع بدايته الى التحول من نظم البرمجة التقليدية بعد الحرب العالمية الثانية إلى إستحداث برامج للحاسبات تتسم بمحاكاة الذكاء الإنسانى فى إجراء الالعب و وضع الحلول لبعض الالغاز والتي أدت بدورها إلى نظم اكبر للمحاكاة، والتي تبلورت بعد ذلك وأصبحت نظما للذكاء الاصطناعى.

فى البداية إختلفت نظرة كثير من العلماء إلى تفسير نظم الذكاء الاصطناعى وإعتبرها بعضهم كفرع من التصميم الهندسى وإعتبرها البعض الاخر بأنها مرتبطة بعلوم محاكاة نظم التفكير الإنسانى، وفى الحقيقة فإن الذكاء الاصطناعى ما هو إلا محاكاة لطرق ذكاء الإنسان ومحاكاة لكيفية إستخدام خبرته المكتسبة فى مجال معين وكذلك طرق تفهمه للغات المختلفة وكيفية التعرف على الصور والتحدث والتي أدت إلى تطور وظهور تقنيات لتصميم برامج تحول الحاسبات الى آلات ذات ذكاء مصنع او لتعمل اعمالا تتسم بالذكاء والخبرة الانسانية.

(١-٢) العقل الإنسانى ومعالجة المعرفة

(Human Mind and Knowledge Processing)

مما لا شك فيه أن عملية معالجة الانسان للمعارف من العمليات المعقدة جدا و يكفى أن نعرف ان قدرة الله سبحانه و تعالى فى ابداع الخلق من ان خلايا العين البشرية تكون قادرة على إدخال ومعالجة و تحليل ما يوازي ٥٠٠ معادلة رياضية تفاضلية لاختطية ولحظية وذلك فى زمن قدره جزء من عشرين جزءاً من الثانية الواحدة . و بمقارنة هذه القدرة فى التحليل بما يقوم به الحاسب السوبر من عائلة كراى (Cray Family) والذي يعتبر من أضخم وأكبر وأسرع الحاسبات السوبر الرقمية فإن ذلك يتطلب عدة دقائق لتحليل مثل هذا العدد من المعادلات وإذا فرضنا أن هناك عينان بشريتان يتبادلان إدخال المعلومات فإنه يمكن القول بان هذا الحاسب السوبر سوف يحتاج لتحليل ما يراه الإنسان فى ثانية واحدة هو اشهر من التشغيل وباستخدام حاسبات الجيل الخامس القادرة على اجراء التحليل الرمزي مثل حاسب الإستدلال المتوازي والتي تبلغ سرعته ألف مليون وحدة إستدلالية فى الثانية فإن هذا الوقت يتم اختصاره بشكل كبير. ومن المعروف أن البيانات الخارجية تدخل إلى المخ عن طريق خمس انواع من الاستشعار مثل الرؤية والسمع والاحساس وغير ذلك حيث يتم تخزين هذه المعلومات والبيانات احتياطيا فى ذاكرة قصيرة الأمد (Short Term Memory) شكل (١-١) وذلك لتحليلها ثم نقلها بعد ذلك الى الذاكرة طويلة الأمد (Long Term Memory) والتي يخزن بها الاشكال والرموز والعلاقات بين هذه الأشكال وكذلك العلاقات التي تستخدم فى المعالجة والتوضيح



شكل (١-١) تمثيل مبسط لعمل المخ البشرى

لأى معلومة فى الذاكرة الاحتياطية، وتعتبر الذاكرة الاساسية التى تحتوى على القليل من البيانات هى الذاكرة قصيرة الأمد والتى تحتاج ما يقرب من ١٥ الى ٢٠ دقيقة لنقل وتخزين هذه البيانات تماما فى الذاكرة طويلة الأمد والدلالة على ذلك إنه إذا تعرض الإنسان لتهتك فى خلايا المخ فى حادثة سيارة مثلاً فإن الذاكرة طويلة الأمد يمكن لها ان ترجع ثانية تماما ولكن ما يحدث فى المدة التى تبلغ من ١٥ إلى ٢٠ دقيقة قبل الحادث فإنه معرض للضياع و فى كثير من الأحوال فإنه لا يمكن إسترجاعه .

ولعقد مقارنة مبسطة بين ذاكرة الحاسب و ذاكرة الإنسان فإنه يمكن إعتبار الذاكرة قصيرة الأمد على انها تماثل ذاكرة التشغيل للحاسب (RAM) والتى تفقد كل ما بها من بيانات إذا تعرض الحاسب لحادثة مثل انقطاع التيار الكهربى، حيث تضيع جميع المعلومات المخزنة بها. أما الذاكرة طويلة الأمد فى مخ الإنسان فهى تماثل الاقراص المغنطة اللينة او القرص الصلب (Hard Disk) حيث تخزن الأشكال والبيانات كمتغير فى نوعية المغناطيسية لمادة القرص او الاقراص المدمجة عالية الكثافة واقراص الليزر (CD) ، وفى الحقيقة فإنه إذا تعرض الإنسان لحادثة سيارة مثلاً والتى يتهتك فيها جزء من المخ فإن مثل هذا الشخص يمكنه ان يشفى تماما و يعود إلى حالته الطبيعية إذا لم تتعرض مراكز الكلام والأعصاب المسئولة عن الحركة إلى التلف وذلك بفضل طريقة التخزين المعتمدة على المعالجة والتحليل والتوزيع المتوازى.

(١-٣) المخ والتمثيل الرمزى للمعرفة

يعتبر المخ الإنسانى وطريقة عمله ووظائفه المختلفة وخصوصا القدرة على المعالجة الرمزية والذاكرة طويلة الأمد من الاشياء الجاذبة لإنتباه كثير من علماء وخبراء الحاسبات الذين يعملون على تطوير الحاسبات الذكية حيث تعتبر بصمات الأشكال والرموز التى تختزن فى ذاكرة الإنسان والمماثلة للبيانات الرقمية التى تخزن فى شبكة المعلومات هى الاساس فى المعالجة الرمزية ومن المنطقى ان يصبح المخ مزوداً بنظام فهرسة رمزى عالى المستوى يسمح بتخزين وإسترجاع هذه الأشكال عند طلبها وذلك بالقيام بتنظيم الاشكال فى مجموعات تحتوى على الحقائق والعلاقات بينهما بشكل يسمح بتخزينها وإسترجاعها كوحدة واحدة، ولقد اثبتت الدراسة أن اكبر عدد من هذه المجموعات التى يمكن أن يعالجها و يوضحها الإنسان تبلغ من ٤ إلى ٧ مجموعات ويمكن اثبات ذلك بالمثال التالى :إذا كتبنا على ورقة الحروف التالية :

ن ل ب ي م ت ن ع ف ك ظ ر ث خ م ر أ و أ ه ن أ ك أ

ثم نظرنا إليها فى مرة واحدة وحاولنا استرجاعها ثانية بدون النظر إليها فإن قدرة الإنسان على الإسترجاع الفورى لهذه الحروف سوف تكون بين ٤ و ٧ حروف متتالية، ويمكن للإنسان التأكد من ذلك بنفسه وقياس قدرته على إسترجاع هذه الحروف، اما إذا كتبناها جملة صحيحة

" نظم الخبرة كثيرا ما تكون نافعة "

وحاولنا إسترجاع هذه الجملة فإننا سوف ننجح فى إسترجاعها بطريقة صحيحة مع أن عدد الحروف متماثل فى الحالتين الا أنه فى الحالة الاولى فإن المخ يحاول ان يتذكر عدد ٢٥ مجموعة ليس بينها اى روابط وعلاقات مخزونة من قبل و لكن فى الحالة الثانية نجد ان المخ يتذكر ٥ مجموعات حرفية و شكلية معروفة للمخ الإنسانى عن طريق الروابط والعلاقات المخزونة من قبل وهذا يفسر الطريقة المعقدة لمعالجة البيانات فى المخ الإنسانى. وتقاس الخبرة للإنسان بكمية المجموعات التى يمكن تخزينها فى ذاكرته بطريقة منتظمة فى دائرة نطاق عمله بشكل طبقي منتظم التفرع بين هذه المجموعات و الذى يظهر فى شكل صفة تصف بها الإنسان من انه ذو تفكير منتظم من عدمه، ولقد قام علماء الحاسبات بوضع مقياس لعدد المجموعات المخزونة فى مخ الإنسان لكى يصبح خبيرا فى موضوع معين والتى تنحصر بين خمسين ألف و مائة ألف مجموعة من الأشكال المرتبطة مع بعضها البعض.

(١-٤) مراحل تطور الذكاء الاصطناعى

يمكن تقسيم الفترات الزمنية لتطور الذكاء الاصطناعى إلى ثلاثة مراحل :

(١-٤-١) المرحلة الاولى

نشأت المرحلة الاولى فور انتهاء الحرب العالمية الثانية وقد بدأها العالم شانون عام ١٩٥٠ ببحثه عن لعبة الشطرنج وإنتهت بالعالم فيجن باووم وفيلد مان (١٩٦٣)، وتميزت هذه المرحلة بإيجاد حلول للألعاب وفك الألغاز بإستخدام الحاسب والتى اعتمدت على الفكرة الاساسية بتطوير طرق البحث فى التمثيل الفراغى الذى يمثل الحالة وأدت إلى تطوير النمذجة الحسابية وإستحداث النماذج الحسابية معتمدة على ثلاثة عوامل هى :

- أ - تمثيل الحالة البدائية للموضوع قيد البحث (مثل لوحة الشطرنج عند بدء اللعب) .
- ب - اختيار شروط إدراك الوصول إلى النهاية (الوصول إلى التغلب على الخصم) .
- ج - مجموعة القواعد التى تحكم حركة اللاعب بتحريك قطع الشطرنج على اللوحة .

ويمكن وضع رسم تخطيطى لهذه العوامل فى الفراغ تمثل فيه الحالات على شكل نقاط (Nodes) وتمثل العمليات (Operations) على انها اقواس انتقال (Arcs) وبذلك يزداد التمثيل الفراغى بين نقاط الالتقاء و الاقواس كلما تقدم اللعب مثلا كما فى لعبة الشطرنج ومن هنا فإن الانتقال من حالة إلى أخرى يتم بين نقاط الالتقاء التى تمثل كل حالة للوحة بعد كل حركة ، ولقد أدت هذه النمذجة إلى إستحداث طريقة تتمثل فى اقتراح الحل وإختبارها وادت إلى سهولة فى وضع الخوارزميات لتمثيل لعبة الشطرنج على الحاسب وفى هذه المرحلة تم تطوير طرق البحث إلى نوعين هما :

١- طرق البحث العمقى " فى اتجاه العمق أولا " .

٢- طرق البحث العرضى " فى اتجاه العرض أولا " .

لتوضيح الفرق بين هذين النوعين من طرق البحث، فإنه يمكن القول ان البحث " فى اتجاه العرض أولا " يتم فى طبقة واحدة الى ان يصل إلى النتيجة المطلوبة، فإذا لم يحدث ذلك فإنه سوف ينتقل إلى الطبقة الأسفل اى ان البحث "فى اتجاه العرض " أولا يحدد اقصر المسارات التى تتبع للوصول إلى النتيجة. اما البحث " فى اتجاه العمق أولا " فإنه ينتقل إلى العمق بسرعة حيث يترك الطبقة التى لا تحتوى على الحل، اى أنه يحدد أقل وقت ممكن للوصول إلى النتيجة وبذلك فإنه يجرى تفضيله فى كثير من الأحوال، كما تميزت هذه الفترة بظهور وتطور البحث الهرمى بإستخدام الحدس (Heuristic Search).

(١-٤-٢) المرحلة الثانية

والتي يطلق عليها المرحلة " الشاعرية " (Romantic) والتي بدأت فى منتصف الستينات إلى منتصف السبعينات، حيث قام العالم منسكى بعمل الإطارات (Frames) لتمثيل المعلومات ووضع العالم ونجراد نظام لفهم الجمل الإنجليزية مثل القصص والمحادثات وقام العالم ونستون والعالم براون بتلخيص كل ما تم تطويره فى معهد الماسيشوسستس للتكنولوجيا والتى تحتوى على بعض الأبحاث عن معالجة اللغات الطبيعية والرؤية بالحاسب والروبوتات (الإنسان الآلى) والمعالجة الشكلية او الرمزية .

(١-٤-٣) المرحلة الثالثة (١٩٧٥-١٩٩٥)

ويطلق عليها (المرحلة الحديثة) والتي بدأت منذ منتصف السبعينات والتي تميزت بظهور التقنيات المختلفة التى تعالج كثير من التطبيقات التى أدت فعلا إلى انتقال جزء كبير من الذكاء الإنسانى إلى برامج الحاسبات، وتعتبر هذه الفترة هى العصر الذهبى لازدهار هذا العلم

والتي ادت إلى ظهور كثير من نظم الذكاء الاصطناعي الحديثة، ولقد تبلورت نواة تقنيات الذكاء الاصطناعي لتشمل النمذجة الرمزية (Symbolic Modelling) وميكانيكيات معالجة القوائم (List Processing Mechanisms) والتقنيات المختلفة للبرمجة (Programming Techniques) والتي تفاعلت مع فروع كثيرة من العلوم كما فى شكل (١-٢) لتشمل الميادين الآتية :

١- اللغات الطبيعية (Natural Languages)

فى هذا المجال ازدهرت فروع اللغويات الحاسوبية (Computational Linguistics) وعلم الفسيولوجى والتعرف والتفهم وتخليق الاصوات والترجمة الآلية والفلسفة.

٢- الرؤية بالحاسب (Computer Vision)

والتي ساهمت فى تطور تقنيات التعرف على البصمات وتطوير الوسائل والتقنيات الالكترونية التى تحاكي نظم الرؤية الطبيعية فى الانسان والتي ساهمت فى تطوير الصناعات المدنية والحربية وكذلك تطور علم الفسيولوجى.

٣- علم الروبوتات (الانسنة) (Robotics)

والتي دفعت فروع الهندسة الميكانيكية والروبوتات الصناعية والتحكم والالكترونيات وعلم السيبرنتيكا الى اغوار تطبيقية بعيدة المدى الاقتصادى والعلمى.

٤- الالعب والمباريات (Game Playing)

ولقد ساهمت الالعب فى تقدم الذكاء الاصطناعي وذلك بادخال ذكاء المستخدم الى البرامج كما ساهمت كذلك فى تطور علوم الحاسب والمباريات الادارية.

٥- اثبات النظريات (Theorem Proving)

والتي ساهمت فى تطور علم الرياضيات وعلوم المنطق وبعض جوانب علم الفلسفة.

٦- نظرية الحاسب والبرمجة الآلية

(Theory of Computation and Automatic Programming)

والتي ساهمت فى تطور علم الرياضيات وعلوم الحاسب.

٧- البحث الهرمى (Heuristic Search)

والتي تشتمل على ميكانيكية البحث وانواعه المختلفة وكذلك تطور النظم الخبيرة.

٨- المكونات المادية للحاسب (Computer Hardware)

والتي ساهمت فى تطور المكونات المادية الالكترونية وتطور علوم الحاسب بشكل عام.

٩- لغات البرمجة والنظم (Programming Languages and Systems)

والتي أثرت علوم الحاسب بلغات جديدة وغلافات تساعد على التخليق لنظم مستحدثة
كما هو مبين فى الابواب القادمة من هذا الكتاب.

١٠- هندسة المعارف (النظم الخبيرة)

(Knowledge Engineering (Expert Systems))

والتي أثرت علوم كثيرة مثل الكيمياء والطب وعلوم الادارة وبحوث العمليات والهندسة
المدينة وصناعة البترول بنظم المعلومات وأدت الى تغيير فى النواحي الاقتصادية وذلك
بتوفير مبالغ كثيرة.

١١- وضع الحل للمشكلات (Problem Solving)

والتي ساهمت فى تطور علم النفس والرياضيات والمنطق.

١٢- تمثيل المعارف (Knowledge Representation)

والتي أدت الى تطور علم الفلسفة وعلوم الحاسب ونظرية النظم.

١٣- النمذجة المعرفية للادراك (Cognitive Modelling)

والتي أثرت كثير من العلوم منها الفلسفة وعلم النفس والمهارات الانسانية والعلوم
العصبية والفسولوجية والموسيقى.

من أهم الانجازات فى هذه الفترة مايلى :

١- النظم الخبيرة المتطورة ومجالاتها المتعددة (Expert Systems)

٢- نظم وبرامج حاسبات الجيل الخامس المشروع اليابانى (١٩٨١-١٩٩٢).

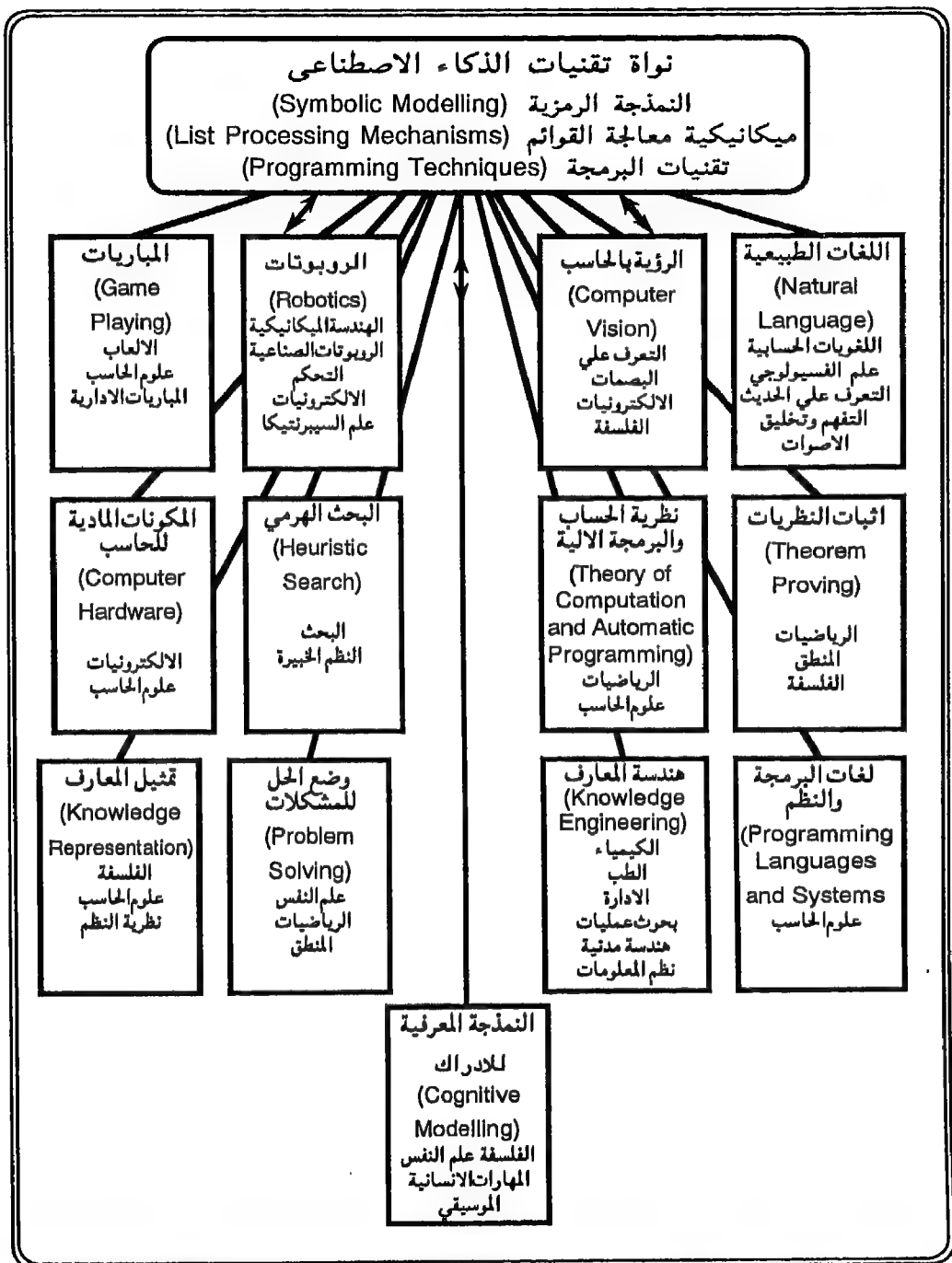
(Fifth Generation Computer Systems)

٣- الشبكات العصبية الاصطناعية والحساب العصبى (١٩٨٠-١٩٩٥م).

(Artificial Neural Networks & Neural Computing)

(١-٤-٤) المرحلة المستقبلية (١٩٩٥-٢٠٢٢م)

يعتقد البعض ان علم الذكاء الاصطناعى مايزال فى مرحلة الطفولة ومن المنتظر ان تتطور
اساليب وتقنيات الذكاء الاصطناعى فى القرن القادم تطورا كبيرا وان تشمل تطبيقات عديدة فى
الحياة العامة لتصل الى اكبر قدر من المستخدمين وقد تمتد هذه الفترة بين سنة ٢٠١٥ وسنة
٢٠٢٥ م وفيما يلى بعض التطورات المنتظرة فى ثلاث فروع من الميادين الأساسية للتطبيقات
الحوية فى هذا المجال.



شكل (١-٢) ميادين العلوم التي صاحبت تطور نواة تقنيات الذكاء الاصطناعي

المجال الأول : اللغات الطبيعية

أ - اقل مراحل التطور المنتظر:

- ١- استحداث نظم لتفهم الحديث المتصل في بعض المجالات الخاصة.
- ٢- الترجمة الفورية للحديث من خلال الشبكات التليفونية لبعض اللغات.
- ٣- التعرف الاوتوماتيكي على النصوص وادخالها فى قواعد البيانات.

ب - مرحلة متوسطة للتطور المنتظر:

تطور نظم التفهم للحديث المتصل والترجمة الفورية والتعرف الاوتوماتيكي للنصوص فى جميع المجالات والذي سوف يتم تدعيمه من خلال قواعد البيانات التى تعمل باللغات الطبيعية والانتاج الرخيص للمكونات الالكترونية للذاكرة وتقدم تقنيات تمثيل المعارف.

ج - أعلى مراحل التطور المنتظر:

يتم الوصول الى وضع الحلول الاساسية لتمثيل المعارف وطرق الاستدلال المختلفة وطرق التعلم وذلك للوصول الى نظم حاسبات تكون لغتها احسن بكثير من بعض المتخصصين وتكون قادرة على الحوار والتفهم والتحدث والترجمة وما الى ذلك.

المجال الثانى : النظم الخبيرة

أ - اقل مراحل التطور المنتظر:

يتم التوسع في توظيف النظم الخبيرة كأدوات لاتخاذ القرار فى ميادين وموضوعات محدودة مع زيادة الاعتماد على مداخل المواهمة التى تعمل باللغات الطبيعية.

ب - مرحلة متوسطة للتطور المنتظر:

فى هذه المرحلة يتم التكامل بين النظم الخبيرة فى شبكات متكاملة فمثلا تتكون الشبكة الطبية من عدة نظم خبيرة يتخصص كل منها فى مجال طبي معين وتصبح الشبكة بيت خبرة عالية المستوى وتترابط الشبكات لتشمل جميع الميادين.

ج - أعلى مراحل التطور المنتظر:

يتم تطوير مايعرف بالالات الذكية والتى تصل الى مستوى ذكاء يتم توليفة من خلال الاتصال المباشر بين الآلة والانسان ويذهب البعض الى القول بان تبادل المعرفة يمكن ان يتم بالاتصال المباشر بين المخ الانسانى ونظام الخبرة وذلك بزراعة نبيطات داخله تكون قادرة على نقل المعرفة الفورية وعند ذلك تصبح الآلة الذكية قادرة على الاداء المماثل للانسان.

المجال الثالث : الروبوتات

أ - أقل مراحل التطور المنتظر:

تتمكن بعض الروبوتات من التحرك ويجرى الاعتماد عليها بشكل مكثف فى وسائل الانتاج المختلفة ولا ينتظر ان تحل مشكلة التحرك والرؤية كاملا وتظل دون المستوى الانسانى.

ب - مرحلة متوسطة للتطور المنتظر:

تعمل الروبوتات فى الحياة العامة مثل التحصيل وقيادة المركبات وفى المنازل.

ج - أعلى مراحل التطور المنتظر:

تصبح حرية التحرك كاملة للروبوتات التى تكون كاملة التجهيز والتى تحمل على متنها الحاسب وكذلك وسائل الرؤية الالكترونية المتكاملة والتى تكون دقتها فى كثير من الاحوال اكثر بكثير من الانسان.

ولتوضيح الانجازات السابقة فإنه يمكن شرح العلاقة بين الإنسان والحاسب فيما يلى.

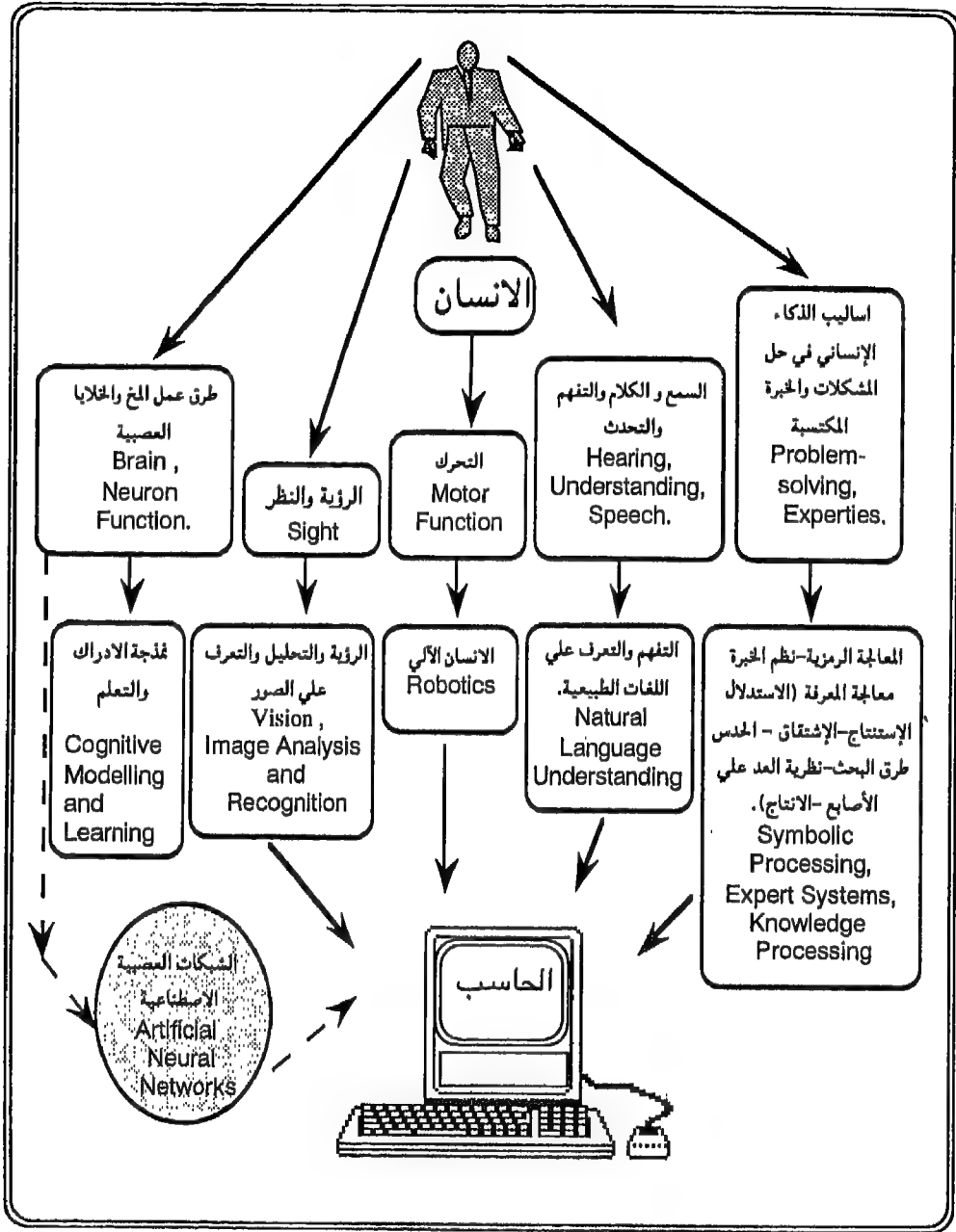
(١-٥) العلاقة بين الذكاء البشرى والذكاء الاصطناعى

يمكن توضيح العلاقة بين الذكاء البشرى (الإنسان) والذكاء الاصطناعى (الحاسب) كما هو مبين فى شكل (١-٣) حيث تم محاكاة ونقل أساليب الذكاء البشرى فى شكل برامج ونظم تجعل الحاسب قادراً على اقتحام مجالات تتسم بالذكاء عند محاولة الحصول على حلول لها وبذلك تم تعريف هذه البرامج والنظم على أنها برامج ونظم الذكاء المنقولة إلى الحاسب او نظم الذكاء الاصطناعى وتتضح العلاقة بين الإنسان والحاسب كالآتى:

١- محاكاة بعض أساليب الذكاء الإنسانى فى موضوعات :

- استخدام الرموز فى التعامل والمعالجة والتعرف على الأشياء.
- وضع الحلول للمشكلات (Problem Solving) واستخدام الخبرات المكتسبة (Experties) للإنسان الخبير فى مجال ما ، ونقلها إلى الحاسب فى شكل برامج ونظم قد أدت إلى نشأة وتطور المعالجة الرمزية (Symbolic Processing) ووضع الحلول للمشكلات. (Problem Solving) ومعالجة المعرفة. (Knowledge Processing) (Expert Systems).

تطورت آليات البرامج التى قائل الطرق المختلفة للتصرف الإنسانى عند تطبيق المنطق مثل ووسائل الإشتقاق (Deduction) والإستدلال (Inference) والإستنتاج والطرق المختلفة للبحث



شكل (١-٣) - العلاقة بين الذكاء البشري والذكاء الاصطناعي
والشبكات العصبية الاصطناعية
(نقل بعض صفات الانسان الى الحاسب لكي يؤدي بعض أعماله)

والموحدات (Unifiers) ومحددات الكمية (Quantifiers) ونظم الإنتاج (Systems) (Production) .

٢ - بمحاكاة أساليب الإدراك السمعى (Hearing) والتفهم (Understanding) والتحدث (Speech) عند الإنسان تم تطوير برامج ونظم التعرف على اللغات الطبيعية وتفهمها ومعالجتها (Natural Language Processing) حيث يقوم الحاسب بتفهم اللغات الطبيعية مثل الإنجليزية واليابانية مثلاً والترجمة الآلية من أحد هذه اللغات إلى الأخرى.

٣ - بمحاكاة أساليب سيطرة المخ والحواس للإنسان على الجهاز الحركى (Motor Function) تم تطوير برامج ونظم الإنسان الآلى وعلم الانسنة (Robotics) وذلك فى محاولة لنقل السيطرة الحركية الدقيقة مع اتخاذ قرار التحرك بناءً على الوضع القائم للاستخدام فى المصانع وما الى ذلك.

٤ - بمحاكاة ونقل نظم الرؤية والنظر للإنسان (Sight) تم تطوير برامج الرؤية بالحاسب (Computer Vision) بمعالجة الصور بطرق مختلفة والتعرف على الأشكال بها (Image Processing and Pattern Recognition) .

٥ - بعمل نماذج لمحاكاة طرق عمل الخلايا العصبية فى المخ (Neurons) وخصوصاً ميكانيكية المعالجة المتوازية أمكن الآتى:

١- وضع نماذج لتصرف العقل البشرى وتطوير علم النمذجة الرياضية لمحاكاة التصرفات الادراكية (Cognitive Modelling) وتطوير نظرية التعلم ومحاكاة طرق المعالجة المتوازية.

٢ - تطوير الشبكات العصبية والحساب العصبى (Artificial Neural Net- works and Neural Computing) والتى تطورت وأصبحت قادرة على محاكاة التعلم والتعرف فى الإنسان. ويمكن القول بأن الحساب العصبى والشبكات العصبية هى محاولة تقليد الإسلوب الذى يتبعه المخ الإنسانى فى العمل، وعلى ذلك فإن الشبكات العصبية لا تعتبر من أحد فروع الذكاء الاصطناعى وذلك لعدم إعتمادها على الأساسيات لهذا العلم، كما انها لا تحمل الخواص العامة له وبذلك جرى تصنيفها على انها مكمل للذكاء الاصطناعى وخصوصاً فى مجالات اكتساب المعرفة والاستدلال والتعلم الآلى. ومن الناحية التطبيقية والإستخدام فإن كثير من تطبيقات الشبكات العصبية لاتدخل فى نطاق هذا العلم والبعض الآخر يندرج تحته ويبين شكل (١-٣) التطبيقات

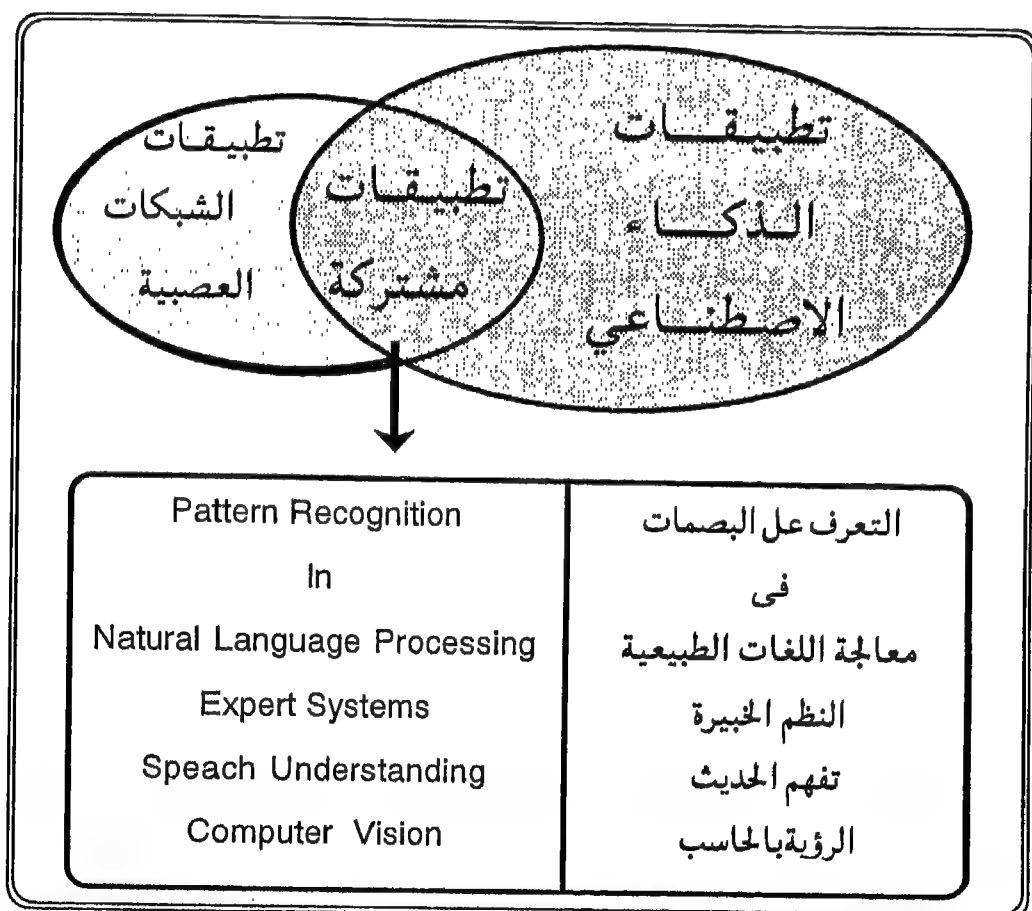
المشتركة بين الذكاء الاصطناعي والشبكات العصبية والتي تشمل التطبيقات المتعلقة بالتعرف على البصمات فى الأفرع المختلفة وبذلك يمكن توصيف الشبكات العصبية كما ذكرنا سابقاً على انها مكمل للذكاء الاصطناعي بغض النظر عن التوصيف العام للشبكات العصبية والذي يختلف فى الجوهر عن التوصيف العام للذكاء الاصطناعي .

(١١-٦) أساسيات نظم الذكاء الاصطناعي

تعتمد دراسة الذكاء الاصطناعي شكل (١-٤) على تفهم الأساسيات الآتية:

- ١- تمثيل المعرفة (Knowledge Representation) : والتي تشمل الطرق الرمزية (Symbolic) لتمثيل المعرفة والتراكيب (Structures) المختلفة المستخدمة فى ذلك والمعانى (Meanings) وكذلك كيفية إكتساب المعارف والخبرات (Knowledge Acquisition) .
- ٢- طرق الاستدلال والتحكم (Inference & Control) : والتي تشمل محاكاة طرق الاستدلال عند الإنسان ودراسة كيف يمكن إستخدام قاعدة التضمين الشرطى المنطقى (إذا توفر الشرط - تكون النتيجة) فى ذلك وإستخدام طرق التحكم المختلفة مثل التسلسل إلى الامام وإلى الخلف.
- ٣ - قابلية التعلم والتكيف (Ability to learn / adapt) : والتي تشتمل على تمثيل قابلية الانسان وكيف يقوم باستخدام الخطأ للتعلم واستخدام دالة الخطأ فى محاولة ضبط القيم الصحيحة وكيف يمكن الوصول الى التكيف (Adaptation)
- ٤ - تمثيل عدم المصادقية او عدم الثقة (او الاستنتاج الغير مكتمل) (Uncertainty Representation) اذا لم تتوفر معلومات كاملة عن موضوع معين او مشكلة ما فان إيجاد الحل يصبح بنسبة معينة من الحل الكامل لهذه المشكلة (Certainty factor) . ولتمثيل ذلك يستعان بالطرق المختلفة التى تعالج عدم الثقة مثل نظرية الاحتمالات (Theory of Probability) وطرق حساب المعاملات ونظرية الدلائل (Evidence Theory) ومنطق فازى (Fuzzy Logic) .
- ٥ - تقنيات البحث والمواءمة (Search & Matching) : والتي تشمل الطرق المختلفة للبحث مثل البحث العشوائى (Blind Search) او الأعمى والذي ينقسم إلى قسمين

يمثل الجزء الاول شمولية تنفيذ البحث (Exhaustive) ويمثل الجزء الثانى جزئية تنفيذ البحث (Partial Search) وذلك بالبحث فى اتجاه العمق اولا (Depth-First) او البحث فى اتجاه العرض اولا (Breadth-Frist) ، كما يمكن محاكاة الإنسان عندما يستخدم حدسه (Heuristics) فى البحث عن حل لمشكلة معينة والذي يتبعه الترتيب الهرمى للمعارف ولذلك يطلق مجازا على هذا النوع من البحث بالبحث الهرمى (Heuristic Search). هذا بخلاف الطرق الرياضية والتحليلية فى إيجاد الحل الأمثل (Optimal Solution) وتشتمل عملية المواءمة على البحث عن الأجزاء المتوائمة فى قواعد التضمين الشرطى.



شكل (١-٣) - العلاقة المشتركة بين الذكاء الاصطناعي والشبكات العصبية الاصطناعية

- ٦- التوحيد والإثبات التحليلي (Unification & Resoluion) : تعتبر عملية التوحيد (Unification) كما سنرى بعد ذلك الأساس فى تصميم ميكانيكية الاستدلال (Inference Mechanism) وتعرف هذه العملية بانها محاولة إيجاد القيم المناسبة للمتغيرات التى تجعل تعبيرين متساويين، كما يعرف الإثبات التحليلي (Resoluion) بانه إستخراج بنود او تعبيرات جديدة من بنود أولية.
- ٧- الاستنتاج المتغير الوتيرة (Nonmonotonic Reasoning) : والذي يعرف بانه إستخلاص النتائج من معلومات او معارف او حقائق عدلت لتناسب الوضع الجديد (والتي تحمل المعنى الموسيقى الذى يشمل الحصول على نغمة جديدة من نغمة معروفة).
- ٨- الوضعية (Empiricism) (التجربة والإختبار) : والتي غالبا ماتسمى توليد التجربة ثم الإختبار (Generat & Test) وذلك بإقتراح الحل الذى يأخذ الشكل الوضعى ثم محاولة إثباته وكثيرا ما يستعان بهذه الطريقة كاسلوب للعمل فى برامج الذكاء الاصطناعى.
- ٩- تفتيت او تجزئة المشاكل (Problem Decomposition) : والتي تعتمد على تفتيت الحل لمشكلة ما حيث يمكن الوصول إلى الحل والذي يتمثل فى اصابة الهدف وذلك بتفتيته إلى مجموعة من الاهداف المصغرة والتي يمكن إثباتها واحداً تلو الآخر.
- ١٠- المشاكل ذات الطبيعة الديناميكية (Problem Dynamics) : والتي تتمثل فى إيجاد الحلول للمشكلات ذات المعارف التى تتغير مع الزمن والتي تستخدم الإطارات (Frames) فى وضع الحلول لها.
- ١١- الأنواع المختلفة للاستنتاج (Types of Reasoning) : وتعتبر الأنواع المختلفة للإستدلال من اهم أدوات التنفيذ لبرامج الذكاء الاصطناعى، ونذكر منها الاستنتاج الاشتقاقي (Deductive) (باستخدام الاشتقاق) والاستنتاج الإستطراذى او الشائيرى او الاستقرائى (Inductive) الاستنتاج بواسطة الاحساس العام (Common Sense) او المشترك.
- ١٢- لغات التمثيل والبرمجة الملائمة للتطبيق (Representation & Pro-gramming Languages) تعتبر لغات التمثيل والبرمجة من أهم أدوات تنفيذ نظم الذكاء الاصطناعى ويتم عرضها بالتفصيل فى الأبواب القادمة.



شكل (١١-٥) الفروض الأساسية لنظم الذكاء الاصطناعي

الفصل الثانى

مجالات

الذكاء الاصطناعى

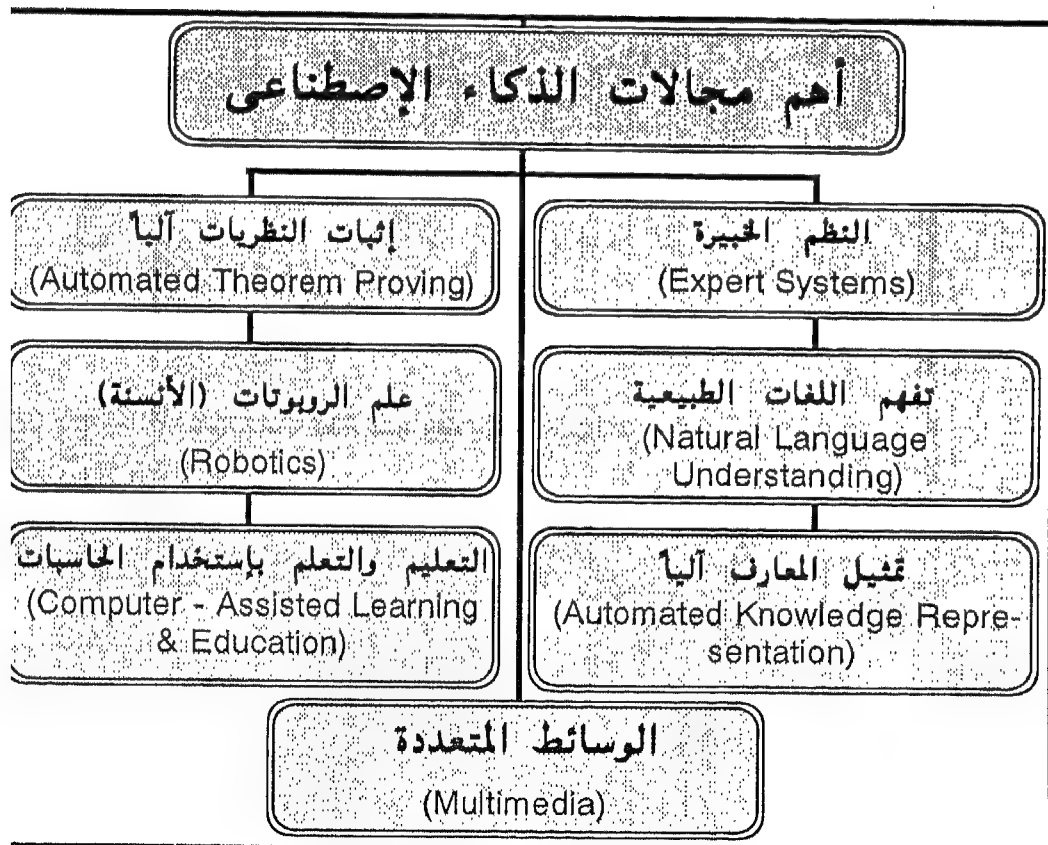
**Domains of
Artificial Intelligence**

(١-٢) مجالات الذكاء الاصطناعي

يبين شكل (١-٢) أهم مجالات الذكاء الاصطناعي كما يلي :

- ١ - النظم الخبيرة (Expert Systems)
- ٢ - اثبات النظريات آلياً (Automatic Theorem Proving)
- ٣ - تفهم اللغات الطبيعية (Natural Language Understanding)
- ٤ - علم الروبوتات (الأنسنة) (Robotics)
- ٥ - تمثيل المعارف آلياً (Automated Knowledge Representation)
- ٦ - التعليم والتعلم باستخدام الحاسبات (Computer - Assisted Learning & Education)
- ٧-الوسائط المتعددة (Multimedia)

ويمكن القاء الضوء على بعض هذه التطبيقات كمايلي:



شكل (١-٢) أهم مجالات التطبيق للذكاء الاصطناعي

(٢-٢) النظم الخبيرة (Expert Systems)

يعتبر إدخال الخبرة المكتسبة للإنسان فى برامج الحاسب فى مجال معين من أهم مجالات الذكاء الاصطناعى وذلك بهدف الوصول إلى برنامج يمكنه أن يعطى النصيحة فى مجال معين أو يحلل البيانات أو الاستشارة أو التشخيص والذى يتطلب استخدام قواعد التضمين المنطقى والتسلسل الأمامى والخلفى إذا تحقق الشرط الحالى فإنه يمكن التقدم إلى الشرط الذى يليه لتحقيقه، وبذلك يتم الحصول على برنامج ذكى للحاسب يستخدم خبرة الإنسان.

ولقد تم استخدام النظم الخبيرة فى كثير من المجالات شكل (٢-٢) أهمها ما يلى:

- ١ - التشخيص الطبى (Medical Diagnosis)
- ٢ - إدارة المعارك الأرضية والطيران (Battlefield & Airforce Management)
- ٣ - التنقيب الجيولوجى (Geological Prospecting)
- ٤ - الصناعات الإلكترونية (Electronic Industry)
- ٥ - الصناعات الكيميائية (Chemical Industry)
- ٦ - التصميم الهندسى (Design)
- ٧ - القانون (Law)
- ٨ - التقنيات الزراعية (Agriculture Technology)
- ٩ - تقنيات الحاسبات (Computer Technology)
- ١٠ - تقنيات نووية (Nuclear Technology)
- ١١ - تخطيط وجدولة الإنتاج (Production Planning & Scheduling)

(٣-٢) إثبات النظريات آلياً (Automated Theorem Proving)

يمكن تعريف الإثبات الآلى للنظريات على أنه تطوير البرامج التى تعتمد على الاستدلال والذى يتمثل فى اشتقاق معادلات جديدة من معادلات معروفة قبل ذلك بحيث تكون المعادلات الجديدة صحيحة فى جميع مجالات المعادلة الأصلية. تعتبر قاعدة التضمين الشرطى الإيجابى (Modus Ponnes) مثال لذلك حيث يمكن إثبات النتيجة التى تنطبق مع الشروط ويمكن للاستدلال أن يتفرع إلى أنماط جديدة وذلك طبقاً للمطلوب اثباته. تعتمد نظرية الإثبات الآلى على نظرية المنطق أحادية الرتبة (First Order Logic) ونظرية الإثبات بالتحليل (Resolution)



شكل (٢-٢) مجالات التطبيق للنظم الخبيرة

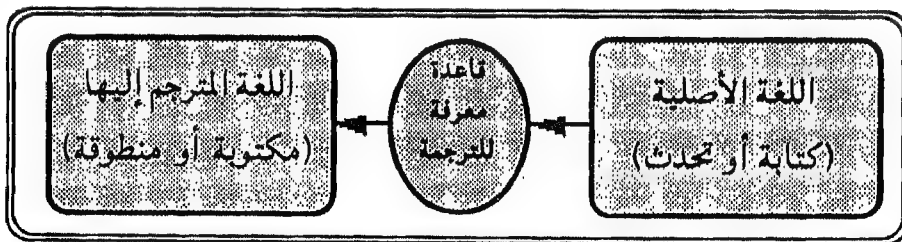
والاثبات بالنفي (Refutation) وغير ذلك من طرق الاثبات المختلفة مثل طرق الاستدلال والإشتقاق المنطقي (Deductive Logic Reasoning) والطرق التحليلية الإجرائية (Procedural Analysis) وطرق إجراء التماثل (Analogy) والإشتقاق بالتأثير

(Deduction By Induction). ويعتبر نظام الخبرة ماكسيما (MACSYMA) من أشهر نظم الخبرة للمعالجة الرياضية الرمزية وكذلك نظام الخبرة للجيل الخامس الياباني (١٩٩٢) للإثبات النظرى الآلى (الآوتوماتيكي) والمسمى " مخلق نماذج الإثبات للنظريات (Model Generation Theorem Provers) (MGTPs) " والذي يقوم بالإشتقاق الآلى (Automatic Deduction) والذي يستخدم قواعد الإستدلال التحليلى عالية الرتبة (Hyperresolution Inference Rules). لقد تم استحداث برامج للإثبات الآلى تقوم بالعمل مثل علماء الرياضيات والتي تشتمل على قدر كبير من قواعد الاستدلال والقادرة على التحوير والاستبدال للرموز بالقيم والمعاني، كما تم استخدام قواعد الإثبات الآلى للنظريات فى تخليق البرامج الآلية بشكل مكثف.

(٢-٤) التفهم والتعرف على اللغات الطبيعية

(Natural Language Understanding)

يتمثل فى بناء وصلات ذكية للمواءمة بين اللغات الطبيعية التى يتكلمها الإنسان ولغات الحاسب (Natural Language Interfaces) التى يتم بها المعالجة داخليا ومحاولة ذلك بهدف ادخال اللغات الطبيعية كطرف بين المستخدم والحاسب مباشرة، ولتصور ذلك فإننا نفرض أن الحاسب يعمل بلغة معينة وانه مثل السائح فى بلد اجنبية يتكلم لغة اخرى فلا بد من وجود مترجم وان هذا المترجم محدود المعرفة فهو يترجم لغة البلاد الطبيعية إلى لغة يفهمها الحاسب، والمطلوب هو تطوير أفق هذا المترجم وتطوير الحاسب بحيث يتم التعامل مباشرة باللغة الطبيعية والتى تختلف عن لغات البرمجة العادية وبين شكل (٢-٣) الأسلوب المباشر للترجمة للحديث او للوثائق المكتوبة والتى تتولى فيه قاعدة معرفة ترجمة الحديث إلى اللغة المطلوبة. نذكر من هذه اللغات لغة " ريتا " والتى تتعامل مع اللغة الانجليزية كلغة للحاسب



شكل (٢-٣) الأسلوب المباشر لترجمة الحديث او النص المكتوب.

وتعتبر أهم الموضوعات التي تقع تحت هذا العنوان هي استخدام الحاسب في الترجمة الآلية (Machine Translation) وإسترجاع المعلومات باللغة الطبيعية (Information Retrieval) والتحدث والتعرف الصوتي بالحاسب وذلك من خلال الشبكات التليفونية وخصوصا ونحن على أعتاب القرن الواحد والعشرين حيث تندمج الشبكات التليفونية وشبكات الحاسبات والشبكات التليفزيونية في شبكة واحدة تعرف بالشبكة الرقمية المتكاملة للخدمات (ISDN) (Integrated Service Digital Network) حيث تتكامل الصورة والصوت والبيانات في شبكة رقمية واحدة .

(١-٤-٢) نظام تفهم للحديث أو الكلام (Speech Understanding System)

يبين شكل (٢-٤) التركيب العام لنظام تفهم للحديث أو الكلام (SUS) والذي يتكون من المراحل الاتية :

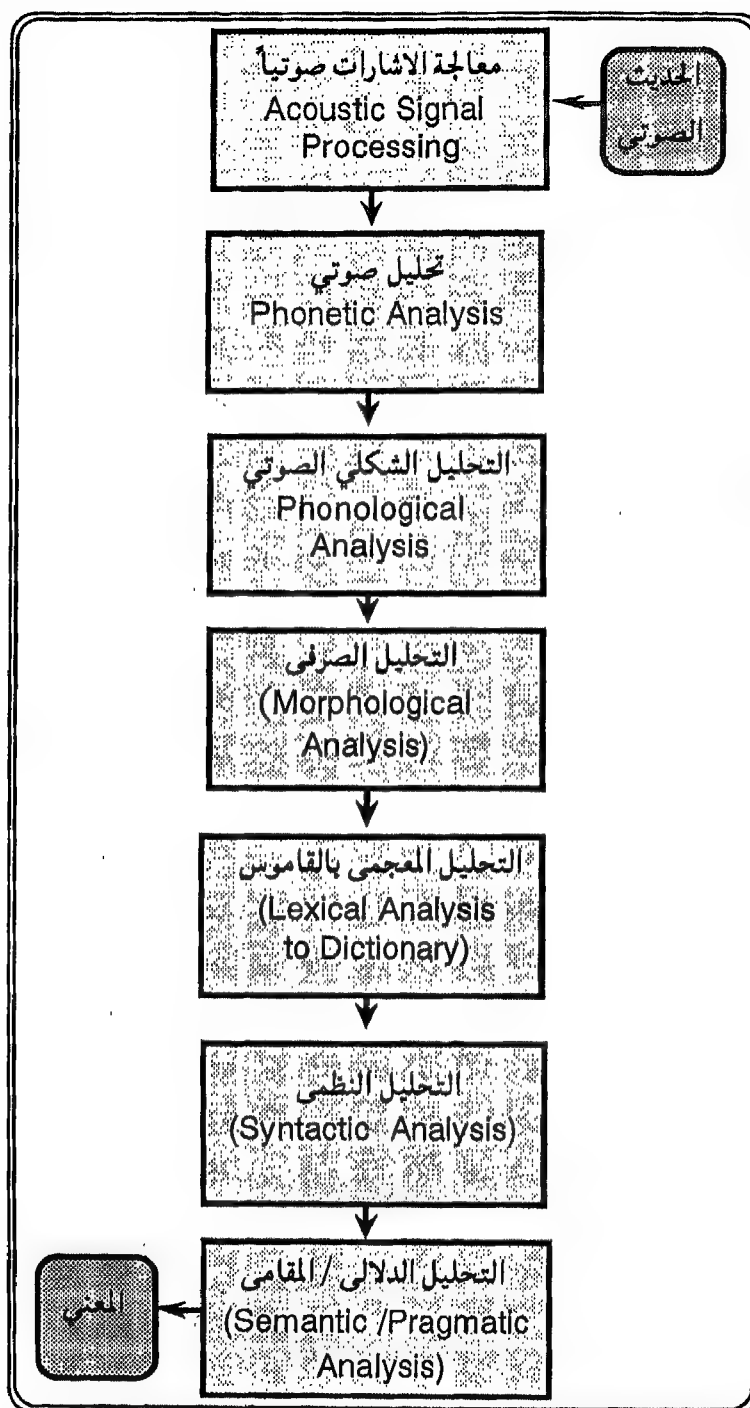
١- مرحلة معالجة الاشارات صوتياً (Acoustic Signal Processing) : يتم تقسيم الحديث المدخّل الي عدد من العينات التي يتم اختيار عددها ثابتا في الثانية ثم يتم تشفير الصوت طبقا للمعرفة المتاحة، وذلك بالتحويل الرقمي للاشارات الصوتية بما يحفظ خواصها الصوتية واستخراج خواص التحليل الطيفي للترددات المحتوية عليها وتحديد التردد الاساسي والتردد المتوسط.

٢- مرحلة التحليل الصوتي (Phonetic Analysis) : يجرى تحديد تبعية الطيف المسجل لاي من الحروف فمثلا اذا كان التحليل الطيفي منخفض السعة ويشبه الضوضاء فهذا يعني ان هذا الجزء ربما يكون جزءا من حرف مثل حرف (ف) واذا كان محتويا علي سعات عالية فهذا يعني انه جزء من حرف (س) مثلا ثم يجري كذلك تحديد المنطوقات و اجزاء الحروف ثم التجميع والضبط للوصول الى التحليل الشكلي الصوتي.

٣- مرحلة التحليل الشكلي الصوتي (Phonological Analysis) : حيث يتم تفسير انواع المنطوقات ومكان الشدة او المد وتحديد اجزاء الكلمات ثم الكلمات .

٤- مرحلة التحليل الصرفي (Morphological Analysis) : يتم تطبيق قواعد الاعراب وقواعد النحو وذلك للوصول الي ترتيب الحروف والمقاطع في الكلمات ثم الكلمات في الجمل .

٥- مرحلة التحليل المعجمي بالقاموس (Lexical Analysis to Dictionary) : يتم تفسير الجمل والكلمات كما بالقاموس الملحق بالنظام ثم يجري التحليل النظمي (Syntactic Analysis) والتحليل الدلالي/ المقامي (Semantic /Pragmatic Analysis) حتى يتم الحصول على معنى الجمل في الحديث.



شكل (٢-٤) مراحل نظام التفهم للحديث (Speech Understanding System)

(٢-٤-٢) الترجمة الآلية للحديث والنص

يبين شكل (٢-٥) الهيكل العام لتقنية الذكاء الاصطناعي للترجمة الآلية للصوت مثل

الحديث مثلا او النص المكتوب والتي تتكون من :

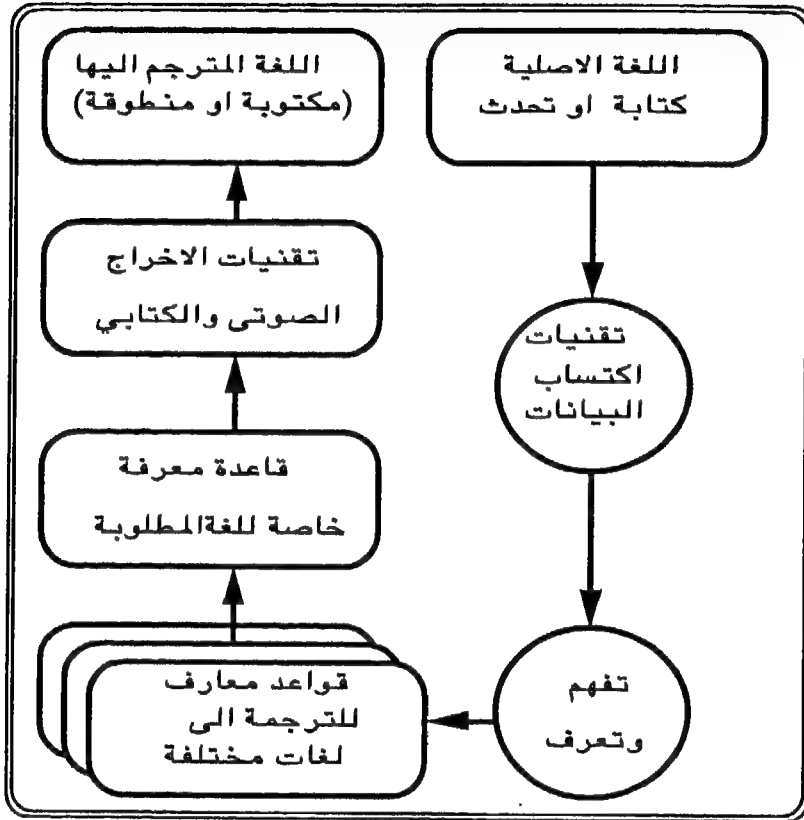
- ١ - الحديث الصوتى المراد ترجمته إلى لغة اخرى او النص المكتوب .
 - ٢ - تقنيات إكتساب بيانات الصوت عند الحديث او إكتساب بيانات النص المكتوب.
 - ٣ - تقنيات التفهم والتعرف والتي يلعب فيها الذكاء الاصطناعي دورا بارزا يتمثل فى التفهم والتعرف على محتويات الرسالة الصوتية باستخدام وسائل التعرف الصوتى او التفهم والتعرف على الحروف والكلمات والجمل فى حالة الترجمة للنص .
 - ٤ - بعد التحديد لمحتويات الرسالة الصوتية او النص المكتوب باللغة الاصلية يجرى إستخدام قاعدة معارف تتولى الترجمة للمفردات والكلمات إلى اللغة المطلوبة.
 - ٥ - حيث أن الترجمة التى تم الحصول عليها ربما لا تحمل المعنى المراد فى اللغة الجديدة لذلك يتم استخدام قاعدة معارف للغة المترجم إليها الحديث او النص لكى تعطى ترجمة المعنى المراد.
 - ٦ - تتولى تقنيات الإخراج الصوتى أو الكتابى اخراج الترجمة للحديث او النص.
- (٢-٤-٣) الأسلوب التتابعى العام للتفهم والتعرف على اللغة

يبين شكل (٢-٦) نموذج للأسلوب التتابعى العام فى التفهم والتعرف على الحديث او

النص والذي يتكون من الآتى :

- ١ - التحليل الصوتى او الشكلى (Phonological Analysis) وذلك باستخدام قواعد المعرفة للتحليل الصوتى او الشكلى وبذلك يتم الحصول على الفونيمات (Phonemes) والجرافيمات (Graphemes) .
 - ٢ - التحليل الصرفى (Morphological Analysis) للفونيمات والجرافيمات للحصول على المورفيمات (Morphemes) وذلك باستخدام قواعد الصرف والمفردات المعجمية.
 - ٣ - التحليل المعجمى (Lexical Analysis) للمورفيمات للحصول على الكلمات وذلك باستخدام المفردات المعجمية.
 - ٤ - التحليل النظمى (Syntactic Analysis) والذي يعمل على تحويل الكلمات إلى تراكيب نظامية وذلك باستخدام قواعد النحو (Grammar) .
 - ٥ - التحليل الدلالى (Semantic Analysis) والذي يشتمل على :
- أ - التحليل المقالى (Discourse Analysis) والذي يستخلص من التراكيب النظامية تراكيب او قوالب منطقية وذلك باستخدام المعرفة المقالية والمعرفة المقامية.

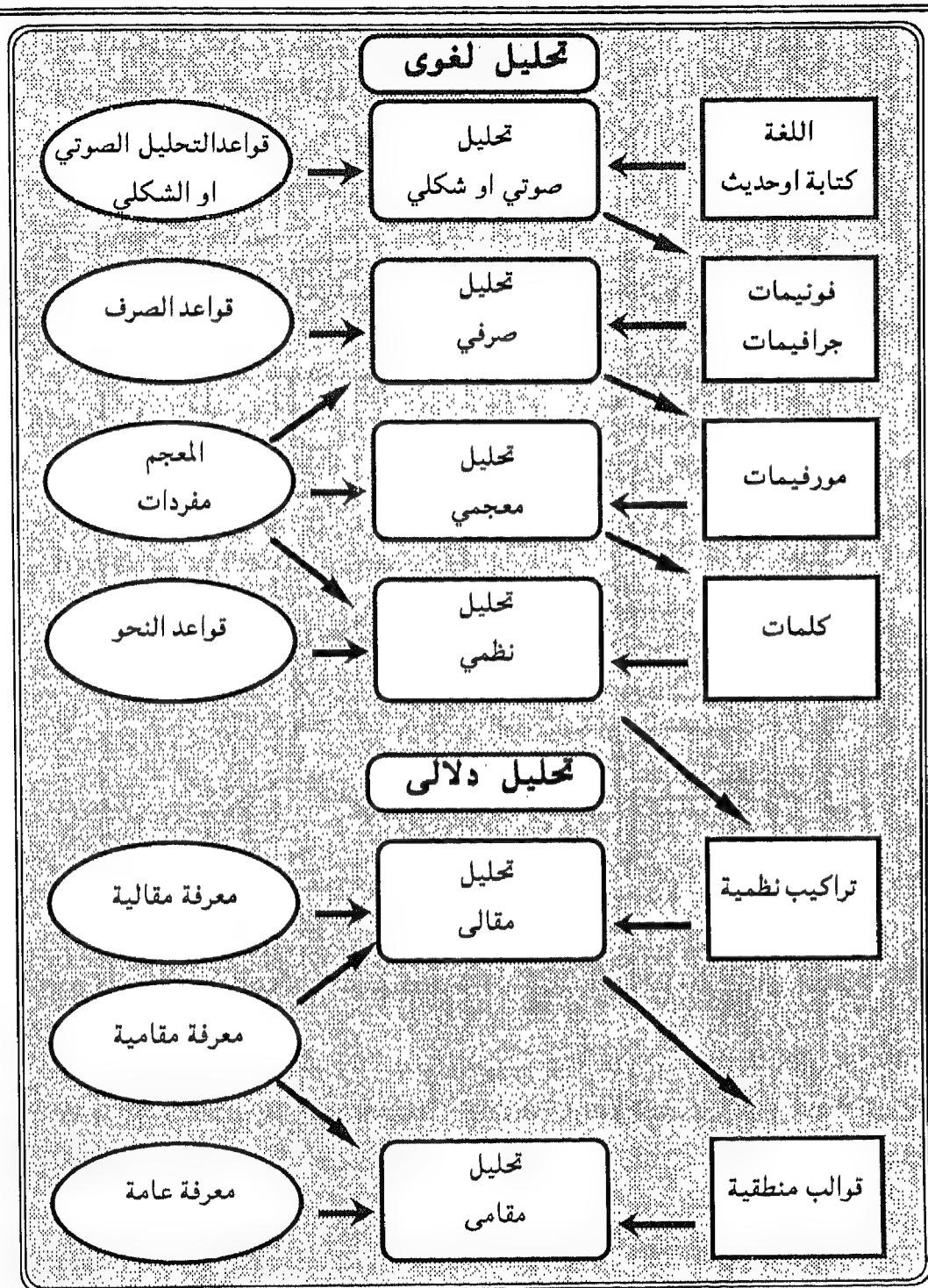
ب - التحليل المقامى (Pragmatic Analysis) والذي يؤدي إلى التفهم الكامل للتركييب والقوالب المنطقية باستخدام المعرفة المقامية والمعرفة العامة وبذلك يتم التفهم والتعرف على الحديث المنطوق او المكتوب. ومن أهم ما تم تطويره فى مشروعات الجيل الخامس اليابانى فى هذا المضمار مايلى :



شكل (٢-٥) الهيكل العام لتقنية الترجمة الآلية للحديث او النص

١ - النظام الترابطى المتوازى لمعالجة اللغات الطبيعية Natural language Processing (Parallel Cooperative System: LAPUTA) والذي يستخدم حاسبات الإستدلال المتوازية (PIM) لإجراء التحليل الصرفى والتحليل النظمى والتحليل الدلائى للغة اليابانية بشكل متوازى .

٢ - مولدات النصوص الاوتوماتيكية (Automatic Text Generation :DULCINEA) والذي يتولى تخليق النصوص طبقاً لمتطلبات المستخدم مثال ذلك المرافعات فى القضايا من قبل المحامين حيث يحدد المحامى الهدف المطلوب ثم يتولى المولد توليد النص الكامل.



شكل (٢-٦) الأسلوب التتابعى العام فى التفهم والتعرف على اللغة

٣ - محلل التركيب المقالى (Discourse Structure Analyzer)

والذى يتولى التحليل المقالى لإستخراج التراكيب والعلاقات المنطقية بين الجمل وإستخراج السمات للمقال بقصد التعرف عليه .

يبين الجدول الآتى بعض نظم التفهم والتعرف المستخدمة علي المستوى التجارى.

بعض نظم التفهم والتعرف للصوت والحديث (Voice & Speech Understanding & Recognition & Systems)

WinVoice	التحكم فى تشغيل النوافذ بالصوت
Voice Type(386& up)	التعرف على ٧٠٠٠ كلمة باستخدام قاموس ٨٠ ألف كلمة
Voice MED vx	نظام صوتي املائي لعمل التقارير الطبية ٥٠ ألف كلمة .
KDS/VOX(PS,OS/2)	نقل المحادثة تليفونيا مع استخدام اللغة الطبيعية وآلة استدلال
Dragon-Dictate 30k	التعرف على اكثر من ٤٠ كلمة فى الدقيقة (٣٠ ألف كلمة) .
Dragon Writer(pc)	التعرف على اكثر من ألف كلمة ويستخدم كادآه للبرمجة

(٤-٤-٤) إنتاج البرمجيات آلياً (Automated Software Generation)

يعتبر الانتاج الآلى للبرامج التي تعمل علي الحاسبات من اهم تطبيقات الذكاء الاصطناعى حيث يصبح الهدف اعفاء المستخدم من التفاصيل المختلفة للغات البرمجة ويصبح التعامل باللغة الطبيعية التي تشفر الي برامج تعمل باحدى لغات الحاسب والتي تؤدي الي برنامج مخلق آلياً. ومن الأمثلة الحديثة لتخليق البرامج آلياً هو استخدام برامج الكتابة علي الحاسب (Word Pro-cessing) لتخليق برنامج لنظم الخبرة مثل برنامج (Icarous MENTOR) والذي يقوم بتوجيه الاسئلة باللغة الطبيعية خطوة بعد الاخرى الى ان يصل الى استيفاء جميع متطلبات المستخدم ثم يقوم بتخليق البرنامج المطلوب.

(٥-٢) علم الروبوتات (الأنسنة) (Robotics)

يمكن تعريف الروبوت او الانسان الآلى كاملى :
الروبوت هو النظام الالى ذو الغرض او المنفعة العامة والذي يشبه الانسان في القيام بأعمال كثيرة ذات ظروف ربما لاتكون معرّفه له سابقا.

ويحتوي نظام الروبوت (Robot System) كما هو مبين فى شكل (٢-٧) من:

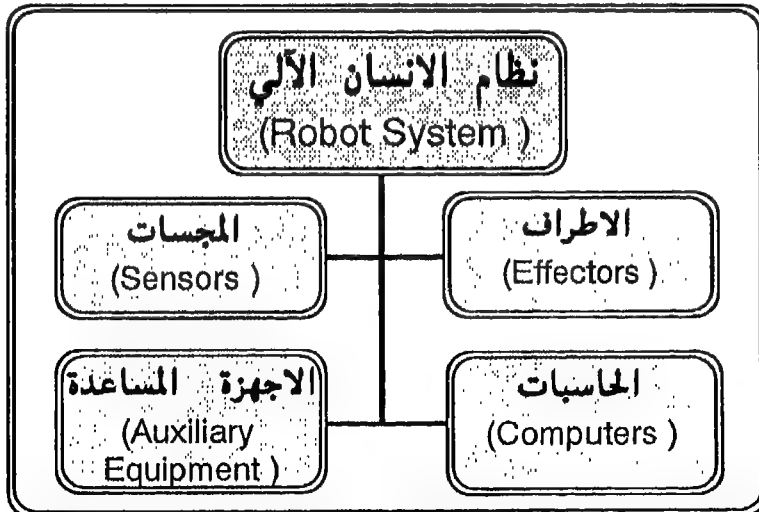
- ١ - اطراف او زوائد (Effectors) والتي تشتمل الاذرع والايادى والارجل والاقدام.
- ٢ - المجسات (Sensors) والتي تشتمل على نقط الاتصال التى تنقل البيانات من البيئة المحيطة الى النظام.
- ٣ - الحاسبات (Computers) والتي تنقسم الى قسمين : الحاسب ذو المستوى الاعلى للتحكم والحاسب ذو المستوى الادنى للتحكم والذي غالبا ما يحتوى على قنوات الاتصال.
- ٤ - الاجهزة المساعدة (Auxiliary Equipment) والتي تشتمل علي الادوات والماسكات والسيور والمحركات وما الى ذلك .

ولقد قام اليابانيون بتقسيم الروبوتات الى الانواع الاتية :

- ١ - روبوت يعمل تبعا لتوجيه الانسان (تابع للانسان) (Slave Robot)
- ٢ - الروبوت ذو خطوة العمل المحدودة (Limited -sequence Robot)
- ٣ - الروبوت ذو الاستجابة للتعليم (Teach -replay Robot)
- ٤ - الروبوت ذو الحاسب المتحكم (Computer - controlled Robot)
- ٥ - الروبوت الذكى (Intelligent Robot)

وتشمل الدوافع لتطوير الانسان الالى الذكى مايلى :

- أ - الدوافع الاجتماعية (Social Incentives) والتي تؤدى الى احلال الانسان الالى محل الانسان العادى ليعمل فى وظائف ذات صبغة مملة او خطيرة مثل الاعمال فى



شكل (٢-٧) مكونات نظام الروبوت (Robot System)

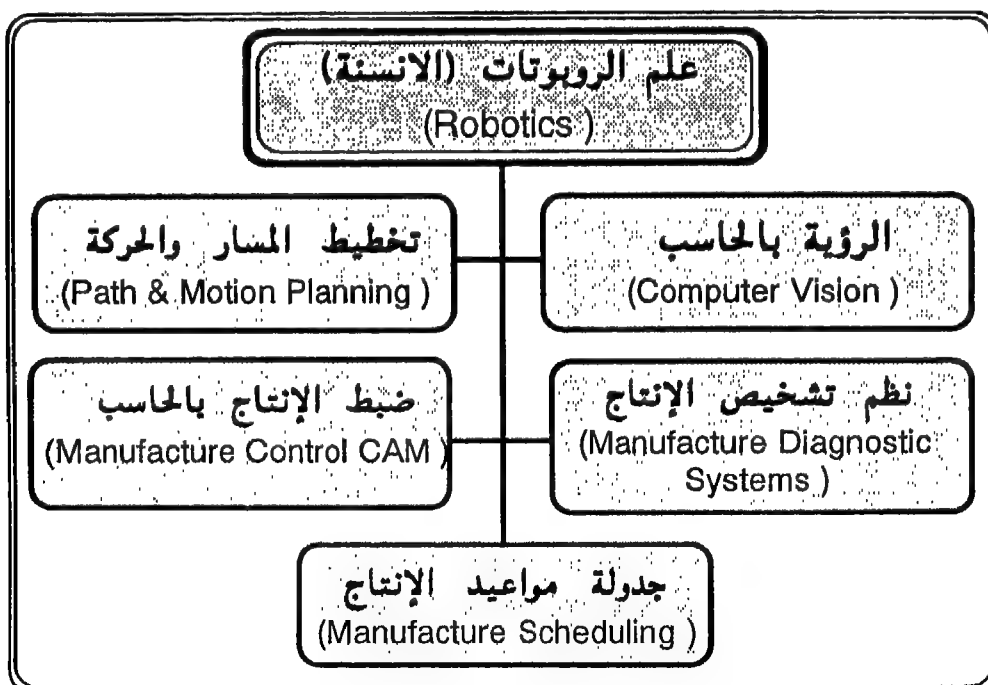
البيئة المحتوية علي اشعاعات عالية و البيئة المحتوية علي ابخرة ضارة والخطيرة مثل مكافحة النيران او التي تتطلب رفع الاثقال او التي تحتوي علي ضوضاء غير محتملة او الاعمال ذات الصبغة المملة مثل التجميع والفرز.

ب - الدوافع الاقتصادية الصناعية (Technoeconomic Incentives) من الواقع ان عدد الروبوتات التي تحل محل الانسان في الصناعة قد ارتفع من ٢٠٠٠ روبوت في العام سنة ١٩٨٠ الى ما يقرب من ٤٠ الف روبوت سنة ١٩٩٠ وبذلك ارتفعت نسبة الاحلال من ٠,٠٣٪ الى ٠,٠٦٪ ومن المتوقع ان تزيد النسبة بشكل كبير بحلول عام ٢٠٠٠.

ويعتبر علم الروبوتات (الأنسنة) من العلوم الحديثة حيث يمكن للآلات باستخدام الحاسبات أن ترى وتحكم على ما تراه ثم تقوم بتنفيذ المطلوب ويشتمل هذا العلم على الموضوعات المبينة في شكل (٢-٨) وهي:

١- الرؤية بالحاسب (Computer Vision)

٢- تخطيط المسار والحركة (Path & Motion Planning)



شكل (٢-٨) الفروع الأساسية لعلم الروبوتات

٣- نظم تشخيص الإنتاج (Manufacture Diagnostic Systems)

٤- ضبط الإنتاج بالحاسب (Manufacture Control CAM)

٥- جدولة مواعيد الانتاج (Manufacture Scheduling)

وباستخدام البرامج الذكية فإن الروبوتات تستطيع التحكم والتحرك كما هو مفروض، وتسير الابحاث العلمية فى هذا العلم فى الأوجه التالية :

١ - تطوير المجسات (Sensors) والكاشفات و خصوصا التى تتصل بالرؤية.

٢ - تطوير نظم البرمجة و النظم الحاسوبية وربطها بنظم التحكم .

٣ - التحكم و حل المشكلات البيئية.

٤ - تطوير برامج الذكاء الإصطناعى التى تتعامل مع حركة الاجسام فى ثلاث أبعاد، وحل المشاكل الناتجة من البيئة المحيطة بها.

(٦-٢) تمثيل المعارف آلياً (Automated Knowledge Representation)

تعتبر قضية تمثيل المعارف من أهم مجالات الذكاء الإصطناعى حيث انها تعتبر حجر الزاوية لجميع النظم الخاصة بها ولالقاء الضوء على ذلك فإننا نبدأ بالاجابة على السؤال التالى:

ما هى المعرفة وكيف يمكن تمثيلها؟

تعرف المعرفة بأنها " التكامل بين الوصف والطريقة والعلاقات " التى ترتبط بهدف من الأهداف او شئ من الاشياء أو كيان من الكيانات (Object) وكثيرا ما يدخل فى توصيفها الوضع الرمزى (Symbolic) والوضع الرقمى (Numeric) كما تعرف المهارة (Skill) على أنها محاولة الوصول إلى التعرف الصحيح على المعارف وإستخدامها بكفاءة للوصول إلى النتيجة المطلوبة و يعتبر المخ الإنسانى مهياً بالفطرة لمعالجة البيانات الشكلية والرمزية حتى فى سن مبكرة، ولكنه لا يقدر ولا يتفوق فى المعالجة الرقمية والتى تقوم بها أبسط الآلات الحاسبة، ومن هنا كان هناك فرق واضح بين المخ الإنسانى والحاسب حيث يتفوق الأول فى الميدان الشكلى أو الرمزى بينما يتفوق الثانى فى الميدان الرقمى.

ويعتبر مجال تمثيل المعرفة هو الإجابة على السؤال الذى يطرح نفسه وهو :

هل يمكن للحاسب أن يقلد المخ الإنسانى فى المعالجة الرمزية للبيانات؟

ويصبح الهدف هو محاولة إيجاد نظرية عامة أو طريقة لتمثيل المعرفة فى أى شكل من

أشكالها لتفى بمتطلبات المعالجة الشكلية او الرمزية للبيانات، حيث تنقسم طرق تمثيل المعرفة إلى ثلاثة أنواع هي :

- أ - الشبكات الدلالية (Semantic Nets)
- ب - الجبر الاعلاني او الإيضاحي (Predicate Calculus)
- ج - نظم تعتمد أساسا على القواعد (Rule Based Systems) .

(٧-٢) التعليم والتعلم باستخدام الحاسبات

(Computer - Assisted Learning & Education)

بدأت تقنيات الحاسبات فى غزو البرامج التعليمية والتدريبية بالجامعات ومراكز التدريب والمدارس منذ فتره كبيره حيث أدت إلى إنتاج برامج تعليمية وتدريبية تناولت الشرح والإبراز للماده العلمية للمقررات والمناهج الدراسية والتدريبية، ولقد دعا ذلك كثير من رجال التعليم والتدريب إلى طرق مفاهيم جديده تدعو إلى زيادة الإعتماد على تقنيات الحاسبات فى التعليم ولقد كان لدخول أساليب الذكاء الاصطناعى كعامل أساسى فى هذه التقنيات الأثر الكبير فى رفع كفاءة العملية التعليمية والتدريبية.

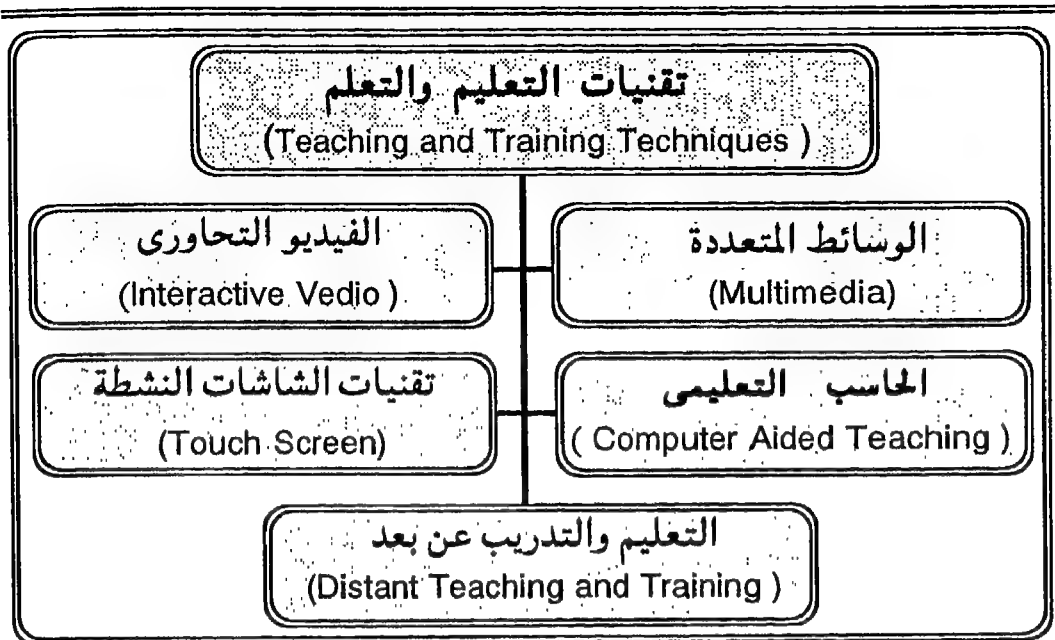
من أهم النتائج التى توصلت إليها كثير من الدراسات والبحوث التى اجريت لتقييم استخدام الحاسبات فى التعليم والتدريب ما يلى :

١ - ان استخدام اساليب الذكاء الاصطناعى والمحاكاة (Simulation) للبرامج العملية والتدريبية قد ساعد على زيادة مهارة الطالب والمتدرب فى الوصول إلى هدف البرنامج التعليمى بسرعه كبيره حيث يمكن إعادة الأجزاء المهمة طبقاً لحاجة المتدرب والذى أدى إلى إطلاق صفة المدرس الذى لا يتعب على هذه النظم.

٢ - الإرتفاع بالمستوى القيادى للطالب عن طريق تعليم نفسه (Self Pacing) وذلك بإتباع الخطوات التحوارية والتعليمية الشارحة للمادة العلمية من خلال البرامج والتقنيات (مثل تقنيات الشاشة النشطة) المعدة لذلك والتدرب على الإختبارات ومعرفة الإجابات الصحيحة مما يؤدى إلى تقييم نفسه ومعرفة مستواه.

٣ - زيادة القدرة الابداعية والتخيلية للطالب وذلك من خلال الاستعانة ببرامج الرسوم الجرافكية والتصويرية (Computer Graphics) فى إبراز المادة العلمية.

٤ - ان ظهور وتطور نظم برامج المؤلف (Authoring Systems) المزودة بالإمكانات



شكل (٢-٩) تقنيات التعليم والتدريب المعتمدة على الحاسب

لإنسياب النص والصورة والمؤثرات الصوتية ساعد على إطلاق إبداع الأساتذة والمدرسين في تكامل الموضوعات وتنبيه معظم حواس الطالب لاستقبال المادة العلمية . وأدى الاستخدام المكثف للحاسب في التعليم إلى ظهور كثير من التطبيقات الحديثة حيث إمتزجت تقنيات الفيديو وتقنيات وسائل التخزين الكبيرة مثل اقراص الليزر والاقراص عالية الكثافة (CD ROM) وكذلك تقنيات التسجيل الصوتي مع تقنيات الحاسب الذي أصبح العمود الفقري لهذه النظم وأدت إلى ظهور التقنيات شكل(٢-٩) الآتية :

١- الوسائط المتعددة لإنتاج الوسائل التعليمية والتدريبية (Multimedia)

٢- الفيديو التحواري (Interactive Vedio)

٣- التدريب بإستخدام الحاسب (Computer Aided Teaching)

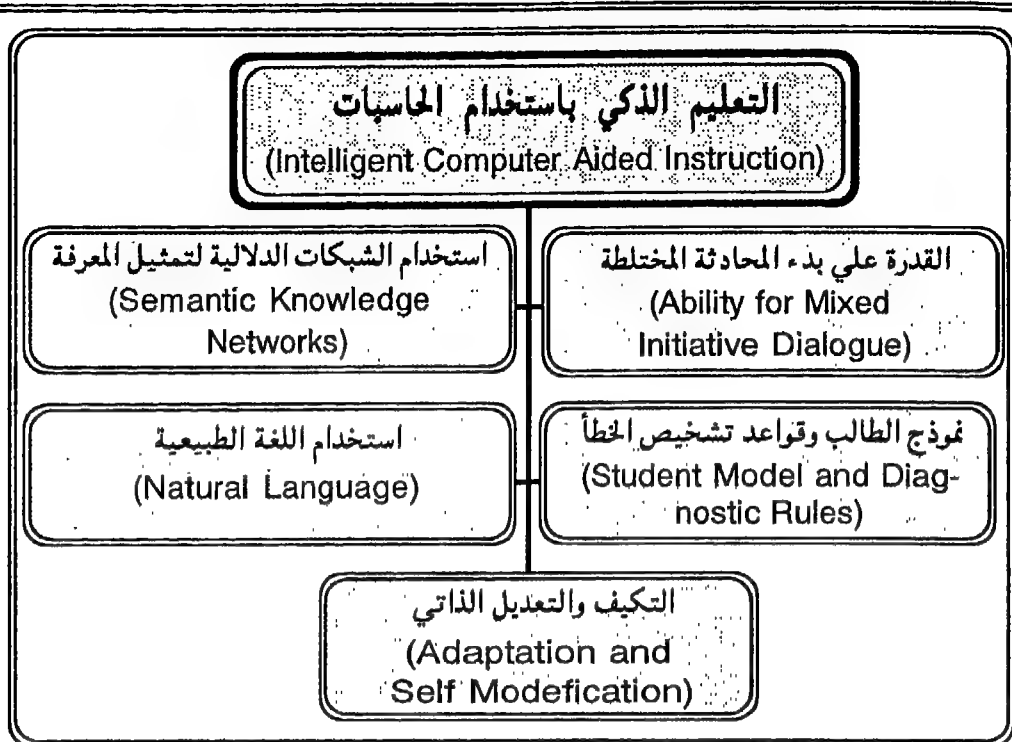
٤- تقنية الشاشات النشطة (Touch Screen) والتدريب التحواري

٥- تقنيات التدريب عن بعد (Distant Teaching and Training)

(٢-٧-١) التعليم الذكي باستخدام الحاسبات

(Intelligent Computer Aided Instruction)

يمكن تعريف التعليم والتدريب باستخدام الذكاء الاصطناعي من خلال تكنولوجيا الحاسبات علي أنه استخدام تقنيات الذكاء الاصطناعي من فروض وبديهيات لانتاج برامج تعليمية



شكل (٢-١٠) الخواص العامة لبرامج التعليم الذكي

وتدريبية قادرة علي التفاعل والتحاور مع الطالب وبيئته. ولقد قام كاربونل (Carbonell) بوضع الأسس والخواص العامة اللازمة لبناء برامج ذكية شكل (٢-١٠) لها السمات التالية :

١- القدرة علي بدء المحادثة المختلطة (Ability for Mixed Initiative Dialogue) والتي تعتبر أهم صفات البرامج التعليمية الذكية والتي تشير الي قدرة البرنامج علي اعطاء الطالب الفرصة لتوجيه السؤال الي البرنامج والحصول علي الاجابة وبذلك يكتمل التحوار بين الطالب والبرنامج في الاتجاهين وذلك علي خلاف ما يحدث في البرامج التعليمية العادية حيث يقوم البرنامج بتوجيه السؤال او المسألة الي الطالب وينتظر الاجابة من الطالب.

٢- استخدام الشبكات الدلالية لتمثيل المعرفة (Semantic Knowledge Networks) حيث تعتبر هذه الخاصية من الاهمية عند تصميم البرنامج حيث يتطلب تمثيل المعارف في شكل شبكات دلالية تحتوي علي الحقائق والقواعد والعلاقات بينهما ويصبح الفرق بين البرامج العادية (CAI) والبرمج الذكية (ICAI) هو قيام المؤلف في البرامج العادية بتنظيم عرض محتويات البرنامج عن طريق التفرع المعرف والثابت وبذلك يحدد شكل وخطوات التحوار، أما في البرامج الذكية فإن القواعد التي استخدمها المؤلف لتحديد شكل التحوار

يجري تعريفها في قاعدة المعارف ثم يقوم البرنامج بتوليد خطوات التحاور وذلك بناءً على نوع الاحابة الصحيحة او الخاطئة للطالب وعلى ذلك فان هذه البرامج تحتوى على نوعين من المعارف الاولى هى المعارف عن موضوع الدرس والثانية هى المعارف عن كيفية تدريس مادة موضوع الدرس.

٣- نموذج الطالب وقواعد تشخيص الخطأ (Student Model and Diagnostic Rules)

من المهم اثناء تشغيل هذه البرامج ان يجري تحديد الخطوة التدريسية التالية (حيث انها غير معرفة مسبقاً) والتي تعتمد على المعرفة الحقيقية لما تعلمه الطالب عند هذه الخطوة، ويمكن الوصول الي ذلك بان يكون البرنامج قادرا على تحديد مستوي تفهم الطالب وذلك بالقيام بتشخيص أية اخطاء يقع فيها الطالب باستخدام وسائل التشخيص المختلفة مثل عدم صحة الفروض والتعميم وعدم القدرة على تقدير صحة المعلومات.

٤- استخدام اللغة الطبيعية (Natural Language) لكى يتمكن الطالب من الاستفادة من

هدف البرنامج لابد من استخدام لغة الطالب الطبيعية حيث تكون المحادثة والتحاور والشبكات الدلالية معبرة بصدق عن المعانى اللازمة لتفهم موضوع الدرس.

٥ - التكيف والتعديل الذاتى (Adaptation and Self Modification) ومن الواجب ان

تكون هذه البرامج قادرة على التكيف مع بيئة الطالب وذلك بالقدرة على تعديل نفسها ذاتيا اعتماداً على ما يدخله الطالب من معارف حقيقية اثناء التعليم وبذلك يزداد حجم المعارف التى توصف بيئة الطالب. وبين الجدول بعض البرامج الذكية للتعليم باستخدام الحاسب.

تعليم الجغرافيا	SCHOLAR
تعليم الارصاد الجوية	WHY
تعليم اكتشاف الاعطال الالكترونية	SOPHIE
تعليم البرمجة بلغة البيسك	BIP
تعليم البرمجة بلغة لوجو	SPADE
تعليم اساسيات البرمجة	FLOW
تعليم الرياضيات في شكل العاب	WEST
تعليم المخاطرة في شكل العاب	WUMPUS
تعليم الحاسب	BUGGY
تعليم التشخيص الطبي	GUIDON
تعليم النظم الخبيرة باستخدام الوسائط المتعددة	KARTT
تقييم مهارات الطالب باستخدام الوسائط المتعددة	HISPANIC

ويعتبر برنامج (SCHOLAR) هو اول برنامج للتعليم باستخدام الذكاء الاصطناعى والذي يتناول تعليم وشرح لبعض الاساسيات بينما يتولى برنامج (WHY) التدريس التحويري للقياسات البيئية للارصاد الجوية وخصوصا تقدير مواعيد هطول الأمطار وتقدير كميتها. ومن أهم البرامج التعليمية برنامج (SOPHIE) لاكتشاف الأعطال والأخطاء فى الدوائر الالكترونية، وترجع الأهمية في ذلك الى القدرة على الاستدلال عن معلومات الطالب العلمية عن الدوائر الالكترونية وأعطالها مع استخدام التغذية العكسية عند الاجابات الخاطئة للطالب لتنشيط وبدء المحادثات. ويقوم برنامج (BIP) والمطور بجامعة ستانفورد بتعليم البرمجة بلغة البيسك حيث يمكنه اقتراح عمل برنامج لمسألة ما ثم يقوم بتصحيح أخطاء البرنامج، وتلي ذلك كثير من البرامج التعليمية للغات البرمجة الاخرى نذكر منها (SPADE, FLOW, MENO-II) كما يعتبر برنامج (WEST) المحاولة الاولى لانتاج البرامج التعليمية التى تستخدم استراتيجيات التدريب بالمراقبة (Coaching) حيث يقوم البرنامج بتقديم النصيحة للطالب عن طريق التحذير واختيار الوقت المناسب لذلك. كما يعتبر برنامج (GUIDON) من البرامج التى وضعت حجر الأساس للبرامج التعليمية الذكية حيث أمكن استخدام قواعد التشخيص الطبي لنظام الخبرة الطبي مايسين لانماء قواعد التحوير والمحادثة مع الطالب حيث يمكن للطالب ان يحدد المستوى المعرفى الذى وصل اليه وكذلك المستوى المعرفى الذى يريد الوصول اليه.

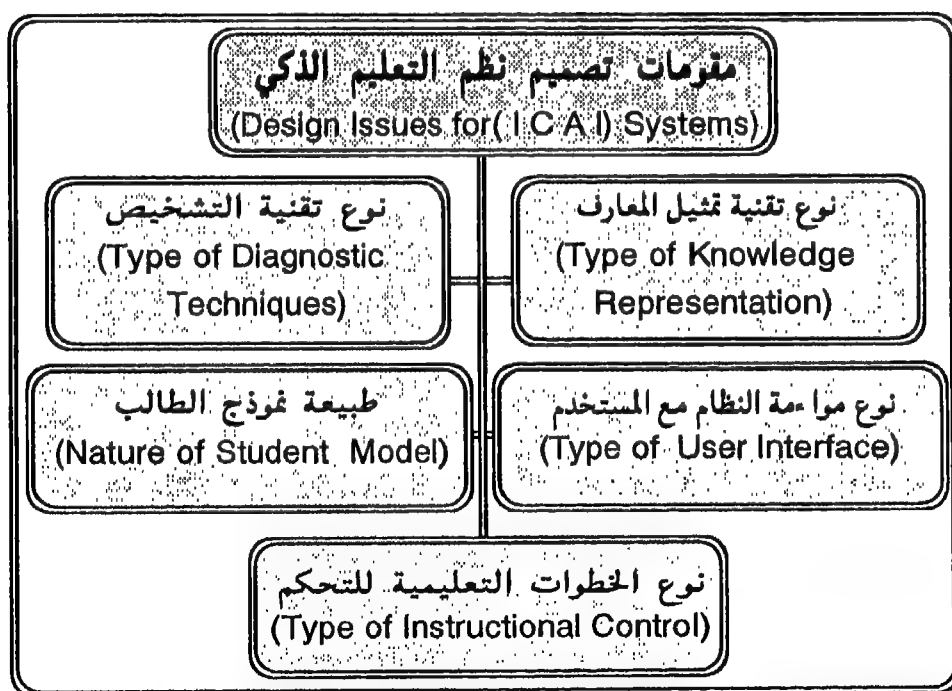
من البرامج الحديثة (١٩٩٢) التى تعتمد علي تقنيات الوسائط المتعددة برنامج (KARTT) وبرنامج (HISPANIC) حيث يعتبر برنامج (KARTT) من أهم البرامج التعليمية والتدريبية لبناء واكتساب الخبرات المعرفية التى تمكن الطالب من بناء نظم الخبرة التى تعتمد علي بيئة الطالب نفسه، ولنا أن نتصور البرنامج في شكل كتاب يحتوى على أجزاء وفصول ويعتمد على الشبكات الدلالية لتوصيف الفروض حيث يجري توصيل المعارف من خلال نقاط التقاء هذه الشبكات كما يسمح باستخدام التنظيم الهرمى للمعلومات حيث يحتوى على أكثر من مائة عقدة شبكية مقسمة الى سبعة عشر ملف، ولقد أدى التنظيم الهرمى الشبكي الدلالى الى حصول الطالب على حرية الحركة داخل معارف البرنامج حيث يسمح باتاحة الفرصة للطالب للتعلم المتعدد، فمثلا يمكنه أن يتعلم تحليل البروتوكولات بينما يتولى عرض جزء من برامج الفيديو الشارحة لهذه البروتوكولات فى نفس الوقت، كما يمكنه مثلا أن يتعلم كيف تتعلم الالة مع ربط ذلك بطرق التعليم المختلفة مثل التعليم المعتمد علي الشرح والتعلم بالتمثيل والاشتقاق الوضعى. كما يعتبر برنامج (HISPANIC) من أهم البرامج التحويرية التى تستخدم الأوساط المتعددة لتعليم وتقييم القدرات الرياضية والحسابية لتلاميذ الصف الثانى عشر (في النظام

الامريكي للتعليم) والذي يحتوي علي تقنية الاستدلال الديناميكي ونظام خبرة تعليمي علاوة علي التكيف العام بالنسبة لتقييم اجابات الطالب.

(٢-٧-٢) مقومات تصميم نظم التعليم الذكي (Design Issues for (I C A I) Systems)
تشمل مقومات التصميم للبرامج التعليمية التي تستخدم الذكاء الاصطناعي كما يبين شكل (١١-٢) العناصر الاتية :

١ - نوع تقنية تمثيل المعارف : يعتبر نوع التقنية المستخدمة لتمثيل المعارف من العوامل الأساسية لتصميم برامج التعليم الذكية ومن أهم هذه التقنيات تقنية النظم المعتمدة علي القواعد (Rule-based) وتقنيات البرمجة الشيئية (Object-oriented) حيث يعتمد تفهم الطالب للدرس علي طريقة عرض وتمثيل الأساسيات والفروض.

٢- نوع تقنية التشخيص : يلعب اختيار نوع تقنية التشخيص دورا مهما في التصميم، ومن التقنيات المستخدمة في هذا الصدد الطرق المختلفة للاستدلال والتفرع الى الأمام والى الخلف وكذلك نظرية اقتراح الفروض واختبار الحل واستخدام الاستدلال الموجه للوصول الى الهدف.



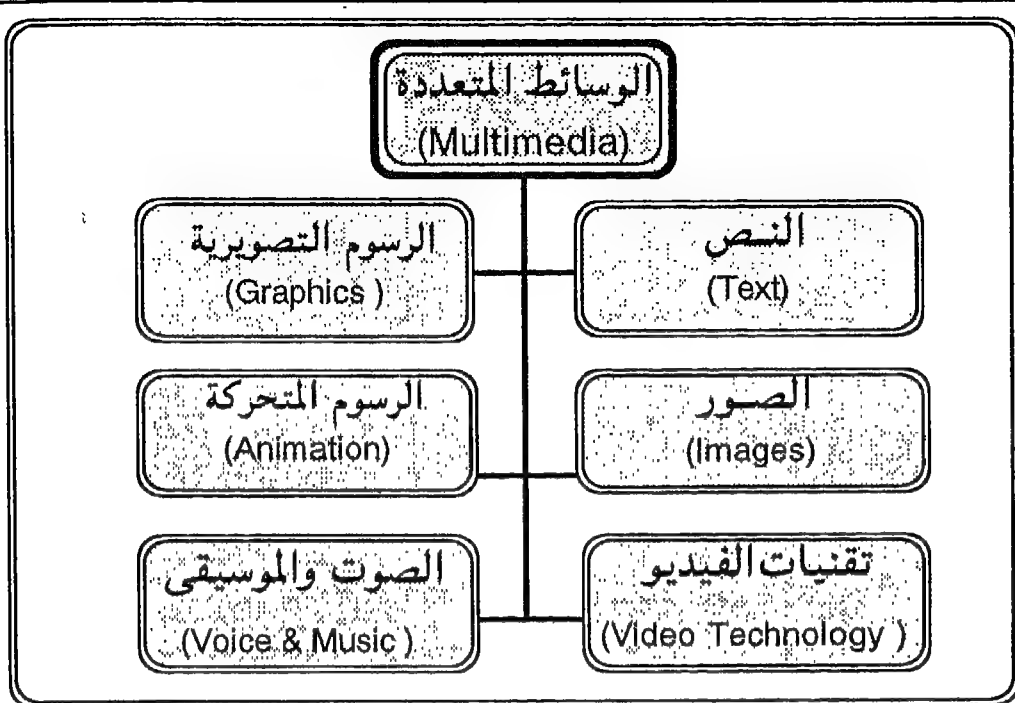
شكل (١١-٢) مقومات تصميم نظم التعليم الذكي

- ٣- **طبيعة نموذج الطالب :** يعتبر نموذج الطالب الواحد هو جزء من نموذج عام يمكنه تمثيل جميع الطلبة والذي يجري تنفيذه في شكل متجه يحتوي علي عدة عناصر حيث يقوم كل عنصر بتمثيل احدي المهارات او احدي الصفات الخاصة التي تعبر عن الطالب، ويعتبر برنامج (Register) مثالا لذلك، بينما يشتمل الاتجاه الاخر بناء نموذج خاص لكل طالب معتمدا علي طريقة تفهم الطالب للفروض التي يطرحها البرنامج.
- ٤- **نوع مواءمة النظام مع المستخدم :** تستخدم معظم برامج التعليم الذكية مداخل للمواءمة مع الطالب باللغات الطبيعية، حيث أن ذلك يتناسب مع بيئة الطالب والموضوع المطروح للتعليم وبذلك يصبح استخدام اللغة الطبيعية أساسيا في هذه البرامج.
- ٥- **نوع الخطوات التعليمية للتحكم :** والتي تعتبر من الأهمية في تنفيذ البرنامج حيث يتطلب بدء المحادثة والتحاور بين الطالب والبرنامج أن تكون خطوات التحكم واضحة وديناميكية والتي تعتمد علي نظرية (ACT) للتعليم والتي ترتبط بالنظم الاجرائية للاتناج وحدود الذاكرة العاملة. ويمكن القول بأن التقدم في تطوير البرامج الذكية للتعليم سوف تزدهر وذلك بتقديم الموضوعات التالية :
- أ - التفهم والتعرف علي الحديث وتخليقه باستخدام الحاسب.
 - ب - تقدم تقنيات الرسوم الجرافيكية والرسوم التصويرية.
 - ج - تقدم أدوات البرمجة في اتجاه برمجة المعارف.

(٢-٨) الوسائط المتعددة (Multimedia)

يمكن تعريف الوسائط المتعددة كما يلي :

الوسائط المتعددة هي التكامل بين مجموعة من التقنيات تساعد على زيادة وإتساع عملية التحوار (Interaction) بين الإنسان والحاسب، وتشمل تقنيات الإدخال والمعالجة والإخراج للنص (Text) والرسوم التصويرية (Graphics) والصور (Images) والرسوم المتحركة (Animation) وتقنيات الفيديو (Video Technology) والصوت والموسيقى (Voice & Music) كما هو مبين في شكل (٢-١٢). ويمكن القول بأن استخدام الوسائط المتعددة قدأضاف الإمكانيات الصوتية والفيديوية إلى عملية التحوار بين الإنسان والحاسب، مثال ذلك استخدام تقنيات الصوت المجسم مع تقنيات الرسوم الجرافيكية والتصويرية لتنفيذ السيمفونية التاسعة لبيتهوفن على الحاسب حيث يمكن النظر إلى أدق التفاصيل للنوتة الموسيقية



شكل (١٢-٢) تقنيات الوسائط المتعددة

والسمع لها فى نفس الوقت.

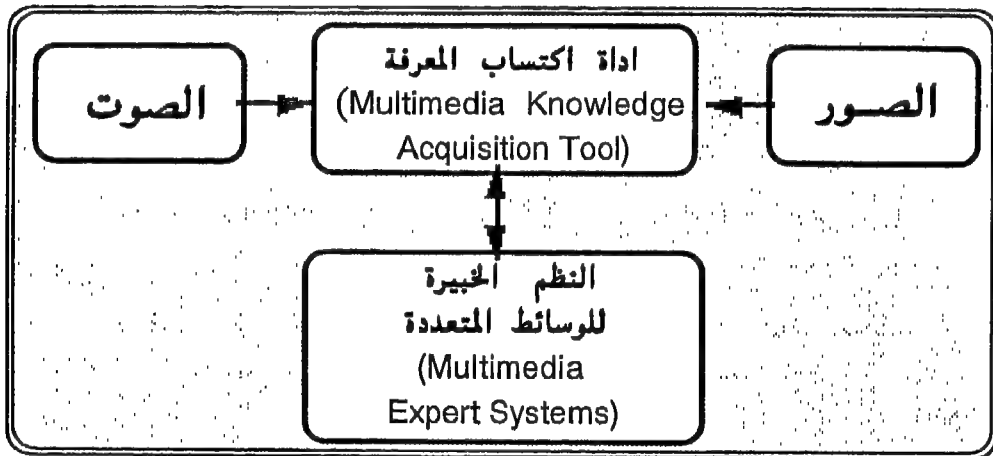
ولإعطاء فكرة للقارئ عن الامكانيات المطلوبة للمكونات المادية (Hardware) اللازمة لإستخدام الوسائط المتعددة على الحاسبات الشخصية والذي يمكن تعريفه بحاسب الوسائط المتعددة الشخصى (Multimedia PC) كما يلى :

- ١ - حاسب شخصى ذو معالج (80486, or, 80586) ذو ذاكرة تشغيل (RAM) ذات سعة (٢) ميجابايت (2MB) على الأقل .
- ٢ - محرك للأقراص عالية الكثافة (CD-ROM) لنقل ١٥٠ كيلوبايت من البيانات فى الثانية.
- ٣ - التعامل مع أى من الثلاثة أشكال للكروت الصوتية وهى (CD, Waveform, MIDI boards).
- ٤ - التعامل مع كارت للفيديو (Video board) والذي يسمح بمشاهدة الإرسال التلفزيونى أو من مسجل الفيديو من خلال نافذة على الشاشة.
- ٥ - استخدام برنامج النوافذ وامكانياته المتعددة (Windows 3.1) .
- ٦ - ونظرا للحجم الهائل للملفات التى تحمل بيانات فيديو فإنه يلزم إستخدام إحدى وسائل ضغط البيانات الفيديوية مثل (VDI) او (JPEG).

تلعب أساليب الذكاء الإصطناعى دورا هاما فى تطوير تقنيات الوسائط المتعددة حيث يمكن اعتبار نظم الوسائط المتعددة مثل نظام الشارع ذو الاتجاهين (Multimedia =Two - Way Street System)

الاتجاه الأول : حيث يعمل الذكاء الإصطناعى على توسيع وتعدد دائرة الإتصال وبذلك يزيد من عرض وإتساع قناة الاتصال للمعلومات التى ترد من التقنيات المختلفة إلى الإنسان.

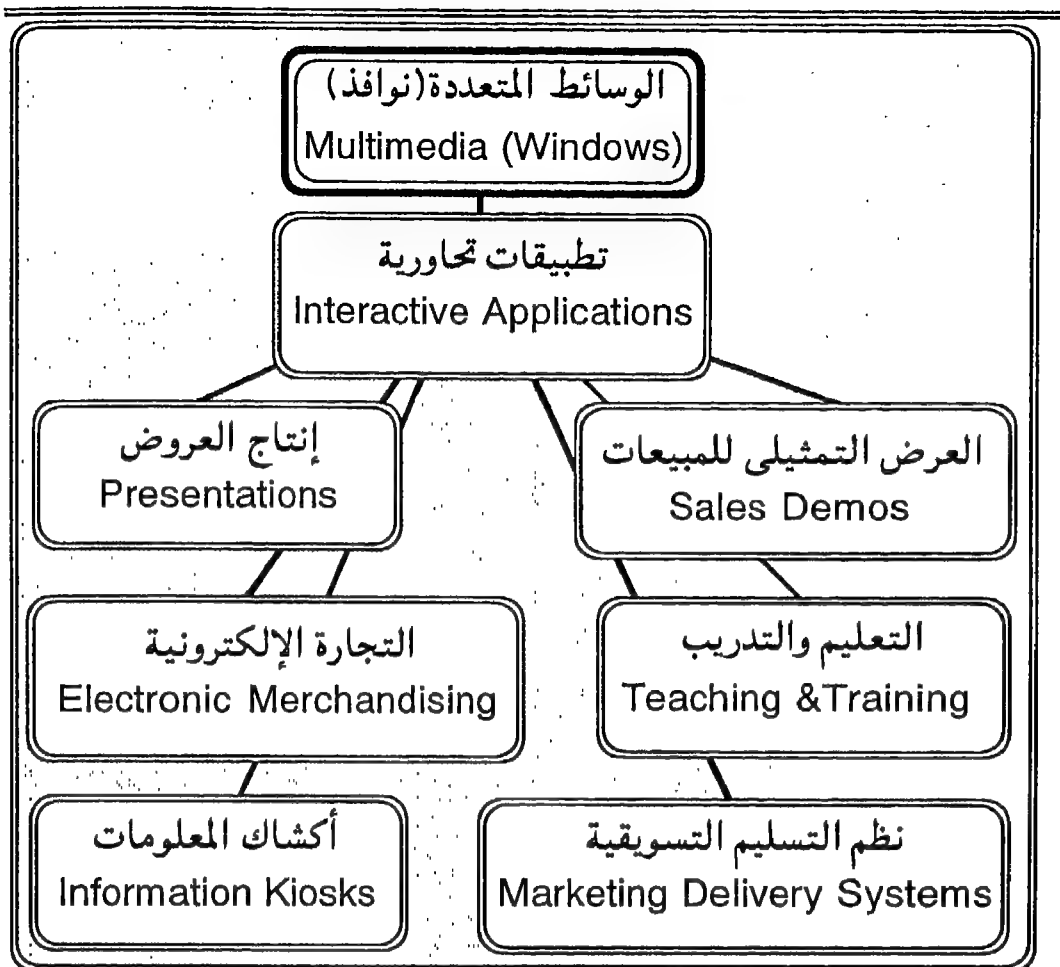
الاتجاه الثانى : او الاتجاه العكسى والذي تعمل فيه الوسائط المتعددة كأداة لاكتساب المعرفة (Multimedia Knowledge Acquisition Tool) والذي يؤدي إلى بناء النظم الخبيرة للوسائط المتعددة (Multimedia Expert Systems) مثال ذلك اذا أدخلنا الصوت لاحدى الماكينات وكذلك صور للقطع المكونة لها فى بناء نظام للخبرة يعتمد علي تصنيف الأصوات فإنه يمكن الحصول على التشخيص للعطب الموجود باحدى الماكينات المعطوبة وكذلك الإقتراحات اللازمة لتغيير قطع الغيار، ويبين شكل (٢-١٣) كيف أمكن تحويل نظم الوسائط المتعددة إلى نظم خبيرة للتشخيص واقتراح الحل.



شكل (٢-١٣) نظام خبير للتشخيص بإستخدام الاصوات

وتوجد تطبيقات كثيرة لتقنيات الوسائط المتعددة (بإستخدام النوافذ) والتي تعتمد على التحوار مع المستخدم حيث تتخذ أساليب الذكاء الإصطناعى كعمود فقرى لبنائها كما هو مبين فى شكل (٢-١٤) ومن هذه التطبيقات ما يلى :

- ١- إنتاج العروض (Presentations).
- ٢- التعليم والتدريب (Teaching & Training).
- ٣- أكشاك المعلومات (Information Kiosks).
- ٤- التجارة الإلكترونية (Electronic Merchandising).
- ٥- العرض التمثيلى للمبيعات (Sales Demos).
- ٦- نظم التسليم التسويقية (Marketing Delivery Systems).



شكل (٢-١٤) بعض التطبيقات التفاعلية للوسائط المتعددة

بعض نظم الذكاء الاصطناعي بالوسائط المتعددة (Multimeadia AI Systems)

Director 3.1	نظام المؤلف التفاعلي مع استخدام الرسوم المتحركة.
Expert Advisor2.3	نظام خبرة لميكنة المكاتب
InterActive	نظام تفاعلي متعدد لإنتاج العروض
Personal HyperBase	برولوج مشفر لإنتاج تطبيقات ذكية.
SELLavision	مجموعة متكاملة لربط نظم الخبرة بالوسائط المتعددة
Quik Point FX150	مجموعة تشغيل ذكية للشاشات النشطة

الفصل الثالث

النظم الخيرة
ومجالاتها المختلفة

**Expert Systems
and its
Different Domains**

(٣-١) النظم الخبيرة (Expert Systems)

يعتبر مجال النظم الخبيرة (نظم المعارف) من أهم المجالات التي تطورت في المرحلة الحديثة والتي ارتبطت باستخدام الحاسبات في إيجاد الحلول للمشكلات التي يتطلب حلها وجود خبرة أو خبير إنسانى في هذا المجال، ولتفهم الدور الأساسى لهذه النظم وكيف يتم ميكنة الخبرة الإنسانية لتدخل ضمن برامج الحاسبات حتي تصبح في النهاية نظاماً برمجية تحتوى على الخبرة الإنسانية في مجال معين وتعطى حكماً أو تصرفاً شبيهاً بالتصرف الإنسانى تجاه حل هذه المشكلات وتنفيذاً لذلك تم الاستعانة بما يلى :

١ - إتخاذ قاعدة التضمين الشرطى المنطقى (IF - THEN) في الشكل :

"إذا توفر الشرط (أ أو الحدث) (أ) يكون العمل أو (النتيجة) (ب)"

كأساس حجر الزاوية في تطوير برامج الذكاء الإصطناعى والتي أطلق عليها وحدة الإستدلال المنطقى (Logical Inference) حيث يستدل على العمل أو النتيجة (ب) إذا توفر الشرط (أ) واختصارها في الشكل (LI) واتخاذها كأساس لتطوير حاسبات الجيل الخامس مثل حاسبات الاستدلال التتابعى والمتوازي.

٢ - يمكن عمل سلاسل تتكون من عدة قواعد بحيث يتطابق العمل أو النتيجة لقاعدة مع الشرط أو الحدث للقاعدة التي تسبقها فإذا عرف الشرط وكان المطلوب النتيجة فإن ذلك يعتبر تسلسلاً في الإتجاه الأمامى (Forward Chaining) وإذا كان العكس هو المطلوب أى أنه تم التعرف على النتيجة أولاً وكان المطلوب تحديد الشرط فإن ذلك يعتبر تسلسلاً في الإتجاه العكسى أو الخلفى أو الراجع (Backward Chaining).

٣ - تم إستخدام قاعدة التضمين الشرطى كأساس في إدخال الخبرة الانسانية على شكل شروط ونتائج وإستخدام التسلسل الأمامى والخلفى لميكنة هذه الخبرة في النظام الخبير.

فما هو النظام الخبير ؟

ويجب على هذا التساؤل الباحث الرئيسى في النظم الخبيرة بجامعة ستانفورد دكتور إدوارد

فاينجن باوم (Edward Feingen Baum) كما يلى:

"نظام المعرفة أو النظام الخبير هو ذلك البرنامج الذكى الذى يستخدم القواعد المأخوذة من الخبرة الإنسانية على هيئة شروط ونتائج في مجال معين وإستخدام طرق الإشتقاق والإستدلال لإستخراج وإستنتاج النتائج المعللة بالأسباب والناجئة عن تطابق هذه الشروط أو النتائج مع شرط أو نتيجة ما

والخاصة بمشكلة معينة يراد إيجاد الحل لها". ويتركب النظام الخبير كما يبين شكل (١-٣) من شقين كما يلي:

أ- الشق الأول : والذي يتم فيه بناء قاعدة المعرفة وذلك بالتسلسل الآتى:

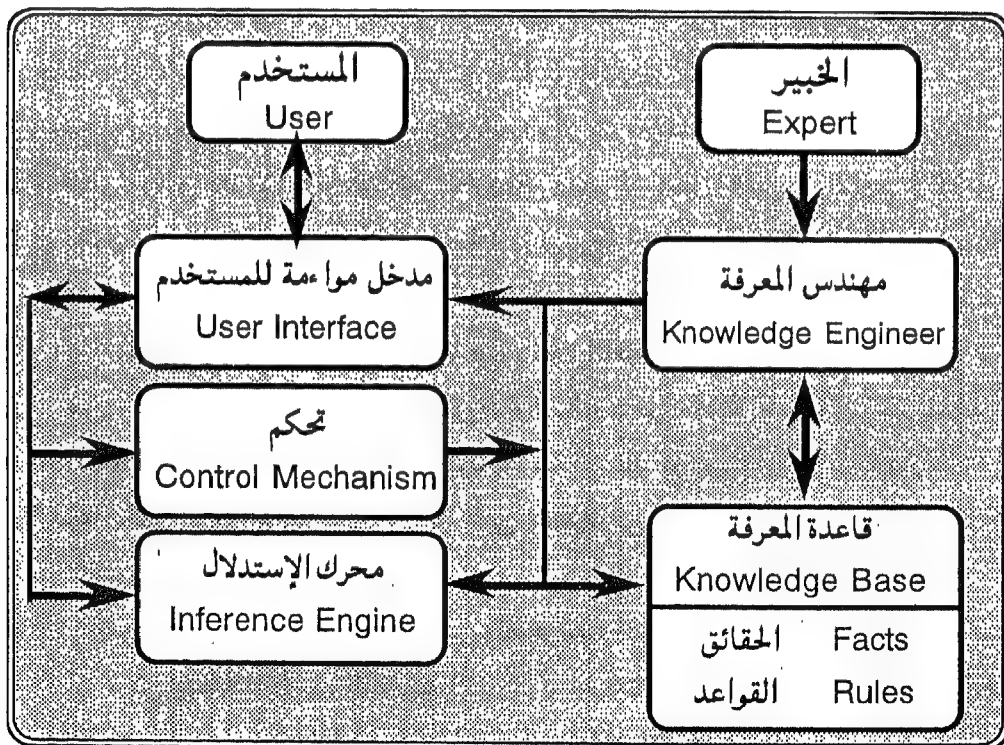
١ - الخبير (Expert) أو مجموعة الخبراء فى المجال المعين المطلوب الحصول على نظام الخبرة فيه، حيث يقومون بإعطاء كل ما جمعه من خبرة بأدق التفاصيل إلى مهندس المعرفة.

٢ - مهندس المعرفة (Knowledge Engineer) والذي يتولى وضع الخبرة فى شكل قواعد للتضمنين الشرطى متضمنة للشروط والنتائج لهذه الشروط وكذلك تطبيق التقنيات المختلفة.

ب- الشق الثانى :والذى يتم فيه طرح الأسئلة من قبل المستخدم والحصول على النتائج أو النصيحة من النظام الخبير والذي يتم بالتسلسل الآتى:

١- المستخدم (User) العديم الخبرة أو ذو الخبرة المحدودة والذي يريد أن يحصل على المعرفة الحقيقية والخبرة المكتسبة لمشكلة ما فى هذا المجال.

٢- مدخل المواءمة للمستخدم (User Interface) والذي يسمح بتبادل المعرفة بين المستخدم



شكل (١-٣) الهيكل الأساسى للنظام الخبير فى مجال معين

وقاعدة المعرفة والذي يجعل سؤال المستخدم مفهوما من قبل قاعدة المعرفة.

٣- أداة التحكم (Control Mechanism) وهى التي تقوم بالتحكم الداخلي في محرك الاستدلال وقاعدة المعرفة للإجابة علي السؤال المطلوب من قبل المستخدم.

٤- محرك الاستدلال (Inference Engine) والذي يقوم بعمل يشبه عمل المحرك وذلك بتوجيه البحث في الاتجاهات المختلفة بقاعدة المعرفة الي أن يتم الإجابة علي سؤال المستخدم.

ويمكن القول بأن تطور علم الذكاء الاصطناعى قد أدى إلى كثير من النظم الخبيرة والتي تعتبر من أهم نتائجه والتي بدأت تغزو التطبيقات المختلفة منذ بداية المرحلة الحديثة إلى الآن (١٩٧٥-١٩٩٥) حيث أصبحت النظم الخبيرة من أهم العوامل المؤثرة إقتصاديا، ويمكن إبراز الفكرة الأساسية للنظم الخبيرة وذلك بعمل محاكاة مبسطة لنظام خبير يعمل كمستشار طبي، ونظام آخر للتعرف علي الصور الجوية ونظام خبير لتحديد الأعطال بالشبكات التليفونية.

(١-١-٣) محاكاة للنظام الخبير " المستشار الطبى "

لكى يؤدي الحاسب دور الطبيب الخبير لتشخيص مرض ما، فإنه يقوم أولا بتوجيه الأسئلة إلى المريض الذى يقوم بدوره بالإجابة على هذه الأسئلة (بنعم) أو (لا) وبذلك فإن الحاسب سوف يطرح الأسئلة على شاشته واحدا تلو الآخر حيث يقوم الطبيب المعالج للمريض (المستخدم) بالإجابة على الأسئلة عن طريق لوحة المفاتيح كما هو مبين فى شكل (٣-٢)، ومن هنا نرى أن البرنامج لكى يحاكي ما فعله الطبيب كان لابد له أن يقوم باستخدام الأداة الشرطية (إذا) و أداة المنطق (و) و أداة الاشتقاق للنتائج (يكون) فى وضع التسلسل الخلفى أو الراجع كالآتى :

تكون (النتائج) الغدد اللعابية منتفخة (و) درجة الحرارة عالية (و) إفراز اللعاب بالفم قليل إذا كان المرض التهاب الغدة النكفية، من الملاحظ أن النتيجة الأولى ليست الخاصة بارتفاع درجة الحرارة ولكنها خاصة بالغدد اللعابية.

من هنا كان لابد من حفظ الشروط والنتائج داخل قاعدة المعلومات بطريقة هرمية (Heuristic) لتساعد على التفرع والوصول بسرعة إلى النتائج حيث أنه لو بدأنا بدرجة الحرارة أولا فربما يكون هناك العديد من الأمراض التي تشترك فى إرتفاع درجة الحرارة، ومن هنا كان ترتيب الشروط من الأهمية لتقليل وقت البرنامج.

ومن الملاحظ أن هذا المستشار الطبى المصغر يستخدم القاعدة الآتية للتضمنين الشرطى :

إذا كان المرض التهاب بالغدد النكفية (الشرط أو الحدث): يكون (الفعل أو النتيجة) الغدد اللعابية منتفخة (و) درجة الحرارة مرتفعة (و) إفراز اللعاب بالفم قليل والعقد الليمفاوية بالرقبة منتفخة (و) يعاني المريض من صعوبة فى البلع.

من هنا يتضح أن هذا النظام الخبير يستخدم التسلسل العكسى أو الخلفى (Backward Chaining)، وبمقارنة المستشار الطبي المصغر مع التركيب المطلوب للنظام الخبير شكل (٣-١) نلاحظ ما يلى :

- ١ - تعاون الخبراء من الأطباء مع مهندس المعرفة لوضع الخبرة المكتسبة في شكل قواعد وحقائق مستخدماً في ذلك قواعد التضمين الشرطى.
- ٢ - تم بناء قاعدة المعرفة والمحتوية علي المعارف اللازمة لتشخيص أحد الأمراض من مجموعة الأمراض المشتركة في كثير من الأعراض ووضعها داخل الحاسب.

٣ - يقوم المستخدم (ذو الخبرة

المحدودة أو عديم الخبرة) بالإجابة علي الأسئلة التي يقوم الحاسب بتوجيهها إلى المستخدم عن طريق شاشته مستخدماً في ذلك لوحة المفاتيح التي تعتبر جزءاً من مدخل المواءمة مع المستخدم وذلك لإدخال الإجابات إلى قاعدة المعرفة باللغة الطبيعية.

٤ - يعمل محرك الاستدلال علي تحديد نوع المرض وذلك بالبحث داخل قواعد المعرفة مستخدماً في ذلك طرق البحث المختلفة وتقنيات التوحيد والطرق المختلفة للإستدلال حتي يعثر علي التشخيص لنوع المرض المقابل للحقائق المدخلة.

٥ - وطبقاً لنوع المرض يمكن إقتراح العلاج.

نظام تشخيص طبي مبسط المستشار الطبي
النظام الخبير: هل الغدد اللعابية منتفخة ؟ المستخدم: نعم
النظام الخبير: هل يعاني المريض من حرارة مرتفعة ؟ المستخدم: نعم
النظام الخبير: هل إفراز اللعاب بالفم قليل ؟ المستخدم: نعم
المستخدم: هل العقد الليمفاوية بالرقبة منتفخة ؟ المستخدم: نعم
النظام الخبير: هل يعاني المريض صعوبة في البلع ؟ المستخدم: نعم
* النتيجة
* نوع المرض : التهاب بالغدد التوكفية

شكل (٣-٢) نظام خبير مبسط في المجال الطبي

(٣-١-٢) محاكاة لنظام خبير مبسط للتعرف الصور الجوية.

مثال آخر للنظم الخبيرة المبسطة هو نظام التعرف على المطارات التجارية من الصور الجوية والذي يتبع تطبيقات الذكاء الاصطناعي في معالجة الصور (Image Processing) والمبين في شكل (٢-٣) والذي يستخدم قاعدة التضمين الشرطى المنطقى الآتية :

" (إذا) توفر الشرط ان (س) = مطاراً تجارياً - (إذا كانت النتائج) (س) صورة جوية (و) تحتوى (س) على ممر أو ممرات (و) تحتوى (س) على طائرة أو طائرات (و) تحتوى (س) على مبنى أو مبان (و) تحتوى (س) على بعض الطائرات تتجه مقدمتها إلى المبنى " .

ويلاحظ استخدام التسلسل العكسى الإتهاء (Backward Chaining) والذي يعطى الشرط إذا توفرت النتائج .

ويعتبر نظام التشخيص الطبى (Mycin) أول نظام خبرة من الناحية العملية حيث قام النظام بتشخيص أمراض الدم الناتجة عن الإصابة بالانواع المختلفة للبكتريا ويتم ذلك عن طريق مقارنة مجموعة الشروط التى تصف بدقة ظواهر المرض البكتيرى المخزونة بالذاكرة فى قاعدة المعرفة الرئيسيه (Basic Knowledge Base) مع مجموعة البيانات التى يحصل عليها الحاسب من المريض عن طريق وصلات المواءمة (Interfaces) التى توصل بارامترات وظواهر المرض إلى الحاسب على شكل أسئلة واجابات تمثل قاعدة البيانات الحقيقية . ولقد قيز هذا النظام فى ذلك الوقت باستخدام قاعدة المعرفة الأساسية على شكل قواعد للتضمين الشرطى :

إذا " - " يكون " (IF - THEN)

وتتوالى بعد ذلك بقية القواعد المؤكدة لظواهر المرض البكتيرى ونوع البكتيريا المسببة له فى تسلسل منطقى خلفى واستخدام الأداة المنطقية "و" (AND) . وبعد التأكد من تطابق نتائج المرض مع نوع الإصابة بالمرض البكتيرى يصل الحاسب إلى إتخاذ قرار لتأكيد نوع الإصابة . كما يعطى هذا النظام نسبة احتمال الإصابة بهذا المرض وذلك بقياس النسبة بين عدد الشروط المتطابقة إلى العدد الكلى للشروط ، فمثلا تبلغ نسبة احتمال التشخيص إلى ١٠٠٪ إذا تحقق التطابق فى كل الشروط بين قاعدة البيانات الحقيقية وقاعدة المعرفة الرئيسيه وتقل نسبة احتمال التأكد بالإصابة إذا لم يتمكن التطابق المتسلسل من الوصول إلى نهاية تلك السلسلة الشرطية ، وقد حقق هذا النظام نجاحا كبيرا فى تشخيص المرض .

من هنا أصبح هذا النظام أول نظام خبرة يودى عملا يماثل عمل الطبيب الخبير فى تشخيص الأمراض البكتيرية وإقتراح العلاج اللازم لذلك وبعد التأكد من تطابق الشروط لنوع الإصابة

نظام خبير مبسط للتعرف على الصور الجوية

النظام الخبير: هل (س) صورة جوية؟

المستخدم: نعم

النظام الخبير: هل تحتوى على ممر أو عمارات؟

المستخدم: نعم

النظام الخبير: هل تحتوى على طائرة أو طائرات؟

المستخدم: نعم

النظام الخبير: هل تحتوى على مبنى أو مباني؟

المستخدم: نعم

النظام الخبير: هل تتجه مقدمة بعض الطائرات إلى المبنى؟

المستخدم: نعم

* النتيجة

* الصورة الجوية (س) تمثل مطاراً تجارياً

شكل (٣-٣) نظام خبير مبسط في مجال التعرف على الصور الجوية

بالمرض البكتيرى يصل الحاسب إلى إتخاذ القرار ولقد استخدم هذا النظام التسلسل العكسى ولغة البرمجة بأسلوب القائمة (LISP).

(٣-١-٣) محاكاة لنظام خبير لتحديد الأعطال بالشبكات التلفونية

(Fault Diagnostic Expert System)

تعتبر النظم الخبيرة للتشخيص من أهم التطبيقات في مجال الإتصالات وخصوصاً تحديد الأعطال، وفيما يلي هيكل نظام خبير لتشخيص الأعطال ودراسة الحركة بالشبكات التلفونية والذي يتكون من جزئين :

الجزء الأول: يتكون من الآتى :

١ - المكونات المادية للشبكة الأصلية (Networking Hardware) والتي تشمل دوائر الإتصال الإلكترونية وملحقاتها.

٢ - وصلة مواءمة تماثلية / رقمية (Analog / Digital Interface) والتي تسمح بتبادل المعلومات والبيانات بين المكونات المادية للشبكة الأصلية وبين برنامج الشبكة والتي يمكن

تبسيطها في حالة إستخدام الشبكات الرقمية.

- ٣ - برنامج الشبكة (Networking Software) والذي يسمح بتحديد العناصر المادية للشبكة والمسارات للخطوط والارقام والعملاء، والذي يتولي ذلك من خلال قاعدة البيانات.
- ٤ - قاعدة بيانات الشبكة (Data Base) والتي تحتوي علي جميع التفاصيل التركيبية للشبكة والتي يمكنها إعطاء بيانات عن الحالة التشغيلية لها وكذلك جميع التفاصيل للعملاء وذلك من خلال برنامج الشبكة والإستعانة بالرسوم الجرافيكية (Graphics) لتوضيح ذلك.

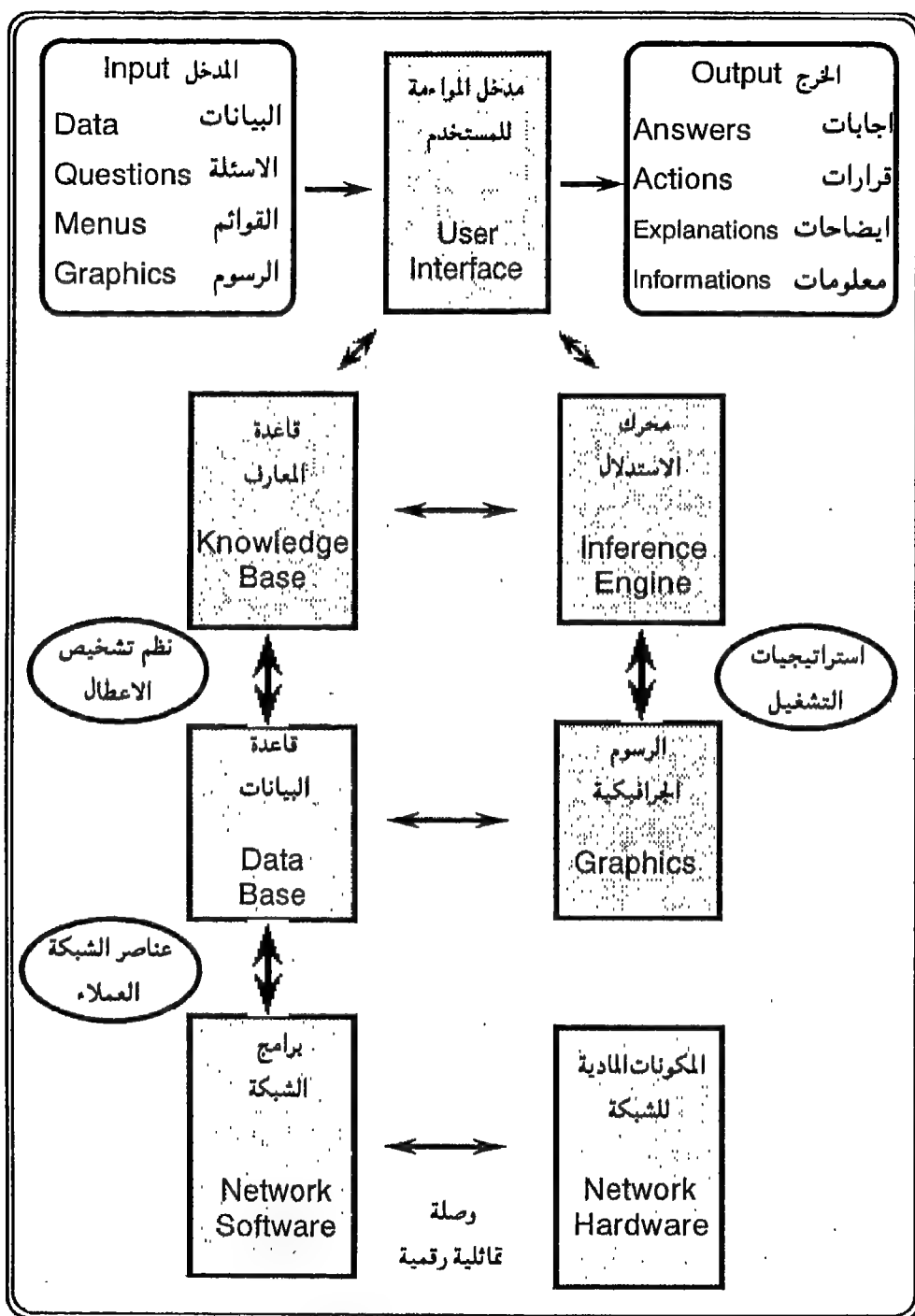
الجزء الثاني : يتكون من الاتى :

- ١ - قاعدة المعارف (Knowledge Base) والتي تحتوي علي جميع الأعطال الممكنة بالشبكة وأسبابها وكيفية إدارة الاصلاح بها وتحتوي كذلك علي نظم تشخيص الأعطال من خلال قاعدة بيانات الشبكة. وتعمل هذه النظم علي تحديد العطل واستخراج بياناته.
- ٢ - محرك الإستدلال (Inference Engine) والذي يقوم بالعمل لتنفيذ ما يطلبه المستخدم عن طريق مدخل المواءمة من إستراتيجيات للتشغيل (Operating Strategies) والخدمات (Services) والتحكم (Control) وتحديد للأهداف (Objectives) والتعرف على الأعطال (Fault Diagnosis) من خلال قاعدة المعرفة وقاعدة بيانات الشبكة والإستعانة بالرسوم الجرافيكية.

- ٣ - مدخل المواءمة للمستخدم (User Interface) والذي يوائم بين متطلبات المستخدم والمعلومات والمعارف الموجودة في قاعدة البيانات وقاعدة المعرفة.
- ٤ - المدخل (Input) والذي يتم عن طريقه إدخال البيانات (Data) وتوجيه الأسئلة للإستفسار عن حالة الأعطال بالشبكة (Questions) وطلب القوائم (Menu) والرسوم الجرافيكية حيث تصل إلى محرك الإستدلال عن طريق مدخل المواءمة للمستخدم.
- ٥ - المخرج (Output) والذي يتم عن طريقه الحصول علي الاجابات علي الأسئلة (Answers) والأفعال المطلوبة (Actions) والإستفسارات (Explanations) والمعلومات (Informations) عن الأعطال وكيفية اصلاحها والبيانات عن الحركة والعملاء والمستخدمين وأى بيانات اخري مطلوبة من قاعدة البيانات وقاعدة المعرفة..

وبين شكل (٣-٤) هيكل نظام الخبرة لتشخيص وتحديد وتحليل الأعطال بالشبكات التليفونية والذي يمكن أن يطبق في حالات مماثلة.

ولايضاح التقدم الكبير فى النظم الخبرة فى منتصف التسعينات فإننا نأخذ النظام الخبر



شكل (٣-٤) هيكل نظام خبير لتحديد الأعطال بالشبكات التليفونية وإقتراح الحل.

الياباني (Model Generation Theorem Provers (MGTPs)) كمثال لهذا التقدم والذي يستخدم فى تخليق نماذج الإثبات الأوتوماتيكي للنظريات الرياضية المرمزة والذي يعتبر من أحد نتائج أبحاث الجيل الخامس (تم الاعلان عنه فى ١٩٩٢م) والذي يجرى إستخدامه على حاسبات الإستدلال المتوازية ((Parallel Inference Machine (PIM)) التى تحتوى على (٢٥٦) من المعالجات المتصلة على التوازي وإستخدام لغة (KLI) حيث يمكن البحث خلال (٣,٠٠٠,٠٠٠) ثلاثة ملايين تفرع إستدلالى فى أقل من أربع ساعات.

(٣-٢) الخصائص والمتطلبات العامة للنظم الخبيرة

- يمكن وضع متطلبات عامة لا بد أن تتوافر فى النظم الخبيرة وهى كما يلى :-
- ١ - أن يحتوى هيكل نظام الخبرة بالدرجة الاولى على قاعدة معارف أساسية تشتمل على شروط مماثلة للخبرة المكتسبة للإنسان وطرق معالجته لموضوع معين لكى يصل فى النهاية إلى التعرف على القرار الصائب كما هو واضح فى نظام التشخيص الطبى السابق ذكره .
 - ٢ - أن يكون هذا النظام قادرا على التعامل مع قاعدة البيانات الكبيرة التى تصف الهدف كما هو الحال فى إستخدام هذه النظم فى التعرف على الصور الجوية أو فى المجال الحربى.
 - ٣ - أن يشتمل نظام الخبرة على أساليب بحث ذات كفاءة عالية نظرا لتعدد قواعد البيانات وقواعد المعرفة وأن يكون النظام قادرا على التفرع السريع.
 - ٤ - أن يكون النظام قادراً على التعامل مع بيانات غير كاملة وناقصة (Incomplete Data) وحتى بيانات مشوشة (Conflictual Data) مثل نظم الخبرة المستخدمة للتعرف على نوع الطائرة من بيانات توضع جزء فقط منها أو إستخدام النظام لقراءة كتابة غير واضحة أو مطموسة .
 - ٥ - إمكانية إدخال بيانات لقواعد المعرفة تحتوى على شروط تمثل خبرات جديدة وذلك لتحديث النظام والوصول إلى ثقة أكبر فى إتخاذ القرار وربما لبناء نظم عملاقة، ومن هنا نرى أن النظم الخبيرة هى التطور الطبيعى لنظم الذكاء الاصطناعى التى تحمل ذكاء وخبرة الإنسان إلى النظم والبرامج المستخدمة علي الحاسبات.

(٣-٢-١) السمات العامة

- من أهم سمات النظم الخبيرة والتى تتميز بها عن بقية النظم الأخرى ما يلى :
- أ - تعتبر قواعد المعرفة قواعد متخصصة فى موضوع أو مجال أو نطاق معين وذلك نظرا

لإختلاف الخبرة بإختلاف المجال، وعلى ذلك فإن نظم المعرفة فى مجال معين لا تنفى بمتطلبات مجال أو نطاق آخر، ومن البديهي أن نظام التشخيص الطبى لا يكون مفيداً فى التعليم أو إيجاد وتحديد الأعطال.

ب - تعتبر طرق الإستدلال والإشتقاق لإستخراج النتائج وهى ما تسمى مجازاً بمحرك الإستدلال أو الحكم (Inference Engine) على الأشياء التى يمكن إستخدامها لأكثر من نظام إذا تغيرت قاعدة المعلومات الثابتة لتناسب النظام الجديد.

ج - لاستخدام طرق الإشتقاق المتعلق بالأسباب لابد من تمثيل جميع القواعد والشروط على شكل مجموعات من الأدوات الشرطية (إذا) توفر الشرط (تكون) النتيجة.

(٣-٢-٢) المجالات المناسبة

تبين الجداول شكل (٣-٥) وشكل (٣-٦) الفرق بين طبيعة المجالات المناسبة للنظم الخبيرة والتى تكون فيها هذه النظم ضرورية بالدرجة الاولى والفرق بينها وبين النظم الحسابية والمحاسبية.

(٣-٢-٣) تجمع النتائج من البيانات (Data Convergence)

من المعروف أن نظم الحاسبات تزداد كفاءتها كلما زادت كميات البيانات والعلاقات عند مخرج الحاسب خصوصاً عندما تكون البيانات عند المدخل صغيرة وتعرف هذه الظاهرة بأنها ظاهرة تفرق البيانات إلى النتائج وعلى العكس، فإن المخ البشرى والنظم الخبيرة تكون أكثر كفاءة

مجالات مناسبة للنظم الخبيرة	مجالات غير مناسبة للنظم الخبيرة
١ - لا توجد حلول وقواعد خوارزمية بل توجد نظم هرمية (HEURISTIC) .	١ - توجد قواعد و حلول خوارزمية (ALGORITHMIC) تناهية تعطى الحل
٢ - يوجد عدد قليل من الخبراء فى المجال	٢ - لا يتطلب وجود خبراء فى هذا المجال ويساوى ذلك وجود خبراء كثيرون من عدمه
٣ - البيانات المتاحة مشوشة (NOISY)	٣ - حقائق دقيقة و معادلات وطرق حسابية
٤ - مجالات تشخيصية أو فى مجال التنبؤ .	٤ - مجالات تستخدم طرق التماثل والإشتقاق العددي لإعطاء الحل .
٥ - المعلومات ثابتة ولا تتغير مع الزمن	٥ - المعلومات والبيانات غير ثابتة

شكل (٣-٥) مقارنة بين مجالات النظم الخبيرة والنظم الحسابية.

النظم المستخدمة فى الحاسبات التقليدية	النظم المستخدمة فى النظم الخبيرة
١ - يحكم التدفق بطريقة منتظمة الخطوات .	١ - يحكم التدفق بالبيانات والمعلومات والمعارف والرموز .
٢ - المعالجة أساساً رقمية .	٢ - المعالجة أساساً رمزية .
٣ - هيكل الحاسب يعتمد على أساس هيكل معمارية فون نيومان .	٣ - هيكل الحاسب لا يعتمد على هيكل معمارية فون نيومان .
٤ - التفرق المعتمد على بيانات قليلة عند المدخل لتعطى بيانات كثيرة عند المخرج .	٤ - التجمع المعتمد على بيانات كثيرة عند المدخل لتعطى نتيجة واحدة عند المخرج .
٥ - المعالجة تسير على خطوات الخوارزميات.	٥ - المعالجة تسير على الخطوات المختلفة لطرق البحث الهرمى .
٦ - يتطلب العمل على الحاسب مبرمج لوضع خطوات الخوارزميات باحدى اللغات.	٦ - يتطلب العمل خبير معرفة لوضع الشروط وإستنتاج النتائج.
٧ - معالجة تتابعية.	٧ - معالجة متوازية ومتوازية.
٨ - الهيكل البنائى يمتد فى شكل خطى كالحظ المستقيم.	٨ - الهيكل البنائى متوازي لاخطى ويمكن ان يصبح دائرى الشكل.

شكل (٦-٣) مقارنة بين طبيعة النظم المستخدمة فى الحاسبات التقليدية والنظم المستخدمة فى النظم الخبيرة

عندما تعطى نتائج قليلة ومحددة عند المخرج من بيانات كثيرة ومعلومات عند المدخل. وتعرف هذه الظاهرة بظاهرة تجميع البيانات إلى نتائج تكون أقل حجماً من البيانات عند المدخل، ومثال ذلك مثل العالم الذى يمضى عدداً من السنوات فى تحليل كثير من المعلومات والبيانات للوصول إلى نتيجة حقيقية واحدة تكون من القوة بحيث أنه ينال عليها جائزة مثلاً. وبذلك يكون النظام الخبير هو ذلك البرنامج الذى يختلف كثيراً عن البرامج العادية المستخدمة فى الحاسبات لأنه يستخدم ظاهرة تجميع البيانات عند المدخل لتعطى بياناً واحداً عند المخرج.

(٣-٣) تقنيات البرمجة للنظم الخبيرة

(Expert System Techniques)

من المعروف أن معظم نظم الذكاء الاصطناعي وكذلك أساليب البرمجة للنظم الخبيرة في مجالات التشخيص الطبي ومجال الرؤية بواسطة الحاسبات واللغات والتصميم والتحكم وكذلك في المجالات الادارية لإتخاذ القرار والتنبؤ والنظم في المجالات الحربية والأمنية مثل التخطيط الحربي ومكافحة الإرهاب الدولي كانت تتطلب العمل علي الحاسبات الكبيرة والمتوسطة في بادئ الأمر مع إستخدام نهايات طرفية، ولقد تطورت البرمجيات والأساليب للنظم الخبيرة لتلائم التقدم الكبير في تكنولوجيا الحاسبات الشخصية من ناحية زيادة سعة الذاكرة وكذلك القدرة التخزينية والسرعة في شكل نماذج أولية (Prototypes) والتي أمكن نشرها بين مستخدمي هذه الحاسبات. وتعتبر تقنيات النظم الخبيرة من تقنيات البرمجة المتطورة، وللإجابة على السؤال الآتي : لماذا تعتبر متطورة ؟ وما هو وجه اختلافها عن البرمجة العادية ؟ فإننا نورد الجدول شكل (٣-٧) الذي

البرمجة العادية	النظم الخبيرة
١- برنامج Program	١- قاعدة المعارف Knowledge Base
٢- المحلل / المنفذ Interpreter / Compiler	٢- محرك الإستدلال Inference Engine
٣- مخطط / محلل برامج Programmer / Analyst	٣- خبير معرفة Knowledge Expert
٤- لغة برمجة عادية FORTRAN, BASIC	٤- أداة برمجة عالية التكوين مثل الغلافة Shell Tool أو أحد اللغات عالية التكوين مثل Rule Talk or ALS Prolog

جدول (٣-٧) - مقارنة بين البرمجة العادية وبرمجة النظم الخبيرة

يبين المقارنة بين اسلوب البرمجة العادية واسلوب البرمجة في النظم الخبيرة.

من الجدول يتضح أن المقابل للبرنامج في النظم الخبيرة هو قاعدة المعلومات وأن آلة الحكم أو الإستدلال التي تشتمل على البحث وتطبيق الشروط في قاعدة المعلومات تقابل المحلل في البرمجة العادية، وأن مخطط البرامج يقابله خبير المعلومات، وأن إستخدام لغة عادية للبرمجة يقابلها في النظم الخبيرة إستخدام أداة برمجة (Programming Tool) عالية التكوين مثل الغلافة (Shell) والتي غالبا ماتكون مستقلة يمكن ربطها بغلافات أخرى، ولتوضيح الفرق بين البرمجة العادية والنظم الخبيرة فانه يمكن القول بأنه في الحالة الأولى يقوم المستخدم عن طريق وصلة مناسبة (لوحة مفاتيح مثلا) بوضع خطوات البرنامج على شكل جمل داخل الحاسب بإستخدام قواعد لغة مناسبة ثم يجري تشفيرها ليقوم المحلل / المنفذ (Interpreter) بتحليلها وتنفيذ خطوات هذا البرنامج ليتعامل مع البيانات التي يتم إدخالها مع البرنامج أو بعد ذلك لاعطاء النتيجة المطلوبة .

ويظهر الفرق في حالة النظم الخبيرة من أن المستخدم يبدأ عن طريق وصلة مناسبة لتحميل قاعدة المعارف التي تختلف كثيرا عن قاعدة البيانات حيث أن الأولى قابلة للحساب (Executable) اما الثانية فهي مجرد بيانات ثم تبدأ بعد ذلك آلة الدلالة أو الإستدلال للبحث و تطبيق الشروط والتفرع إلى الأمام وإلى الخلف في تنفيذ خطوات وشروط قاعدة المعرفة على البيانات التي يتم إدخالها، وعلى ذلك فانه لابد من وجود وسيلة تفاعلية للشرح لكي يتسنى للمستخدم الاجابة عن الأسئلة التي سوف تطرح من قاعدة المعرفة وبذلك يتم إدخال العوامل المتغيرة التي تناسب المستخدم وربطها بقاعدة المعرفة، ثم يأتي دور التعامل مع البيانات لتعطي النتيجة المطلوبة مشتملة على الشروط والقواعد التي طبقت والتي تحتوي كذلك على جزء من خبرة المستخدم .و يمكن إلقاء الضوء على عناصر البرمجة للنظم الخبيرة في شكل ملخص كالآتي :

أ - قاعدة المعرفة التي تعتمد على نوعين من المعلومات:

١ - الحقائق (Facts) .

٢ - القواعد أو الشروط التي تبين العلاقة بين هذه الحقائق.ويمكن تمثيل هذه الحقائق

والشروط بإستخدام التسلسل المنطقي لتنفيذ الشروط بإستخدام قاعدة العد على أصابع اليد (Rule of Thump)، وإستخدام إطارات (Frames) لتمثيل الحقائق والعلاقات بين الحقائق، وإستخدام المنطق (Logic) للتداول والاجابة عن الأسئلة.وعلى ذلك تعتبر الجملة الشرطية: إذا تحققت الشروط : الحقيقة (١)

والحقيقة (٢) والحقيقة (٣) تكون النتيجة الحقيقة (٩) والحقيقة (١٠) وهكذا هي أساس العمل في قاعدة المعرفة.

ب - محرك الاستدلال أو أداة اتخاذ القرار (Inference Engine) التي تشتمل على أسلوب للبحث والحكم والتفرع إلى الأمام وإلى الخلف، وهي التي تقوم بتنفيذ قواعد التضمن الشرطي الموجودة في قاعدة المعارف كالآتي :

إذا تحقق الشرط (١) تكون النتيجة (٢) وإذا تحقق الشرط (٣) تكون النتيجة (٤) وهذا يعني أنه إذا لم يتحقق أى شرط فلا تكون هناك نتيجة، وغالبا ما تزود آلة الحكم هذه أو أداة التنفيذ كما يطلق عليها بمنطق لتطبيق نظرية الاحتمالات في إتخاذ الحكم، وذلك بإتباع نظرية التصديق (Belief Function). ويمكن إدخال الاداة المنطقية (أو) (OR) لتصبح الجملة الشرطية إذا توافر الشرط (١) والشرط (٢) والشرط (٣) تكون النتيجة (٧) أو (٨) وهكذا.

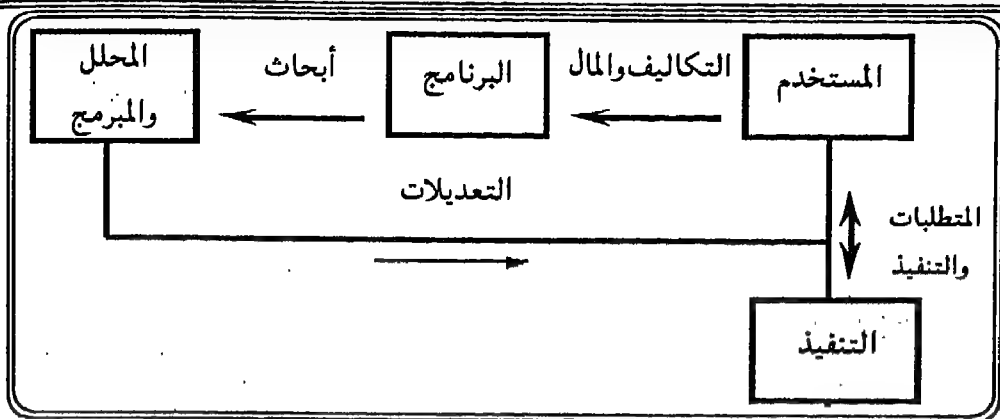
ج - الوصلة بين المستخدم والحاسب وهذه الوصلة تسمح بإدخال المستخدم كطرف في بناء النظام كما في شكل (٣-٨) حيث يجري محاولة إدخال المستخدم كطرف في بناء النظام الخبير من قبل المصمم. وكذلك شكل (٣-٩) حيث أصبح المستخدم طرفا وعنصراً هاماً في بناء النظام الخبير.

(٣-٤) اللغات والحزم المناسبة للنظم الخبيرة

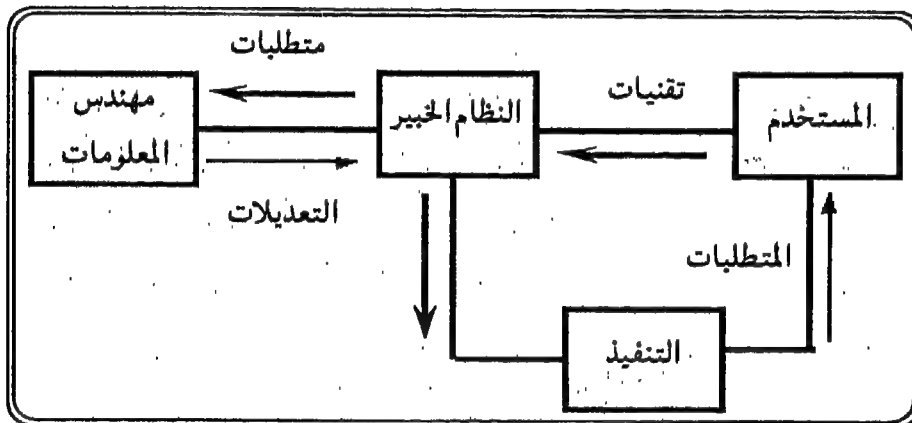
تعتبر الثلاث لغات الآتية ومشتقاتها هي اللغات الاساسية لبناء نظم الذكاء الاصطناعى والنظم الخبيرة على الحاسبات الشخصية وهي :

- ١- لغة البرمجة بإسلوب القائمة - لسب (LISP) ومشتقاتها.
- ٢- لغة البرمجة بإسلوب المنطق - البرولوج (PROLOG) ومشتقاتها.
- ٣- لغة البرمجة بالاهداف أو البرمجة الشيئية - (Object Oriented) ومشتقاتها، مثل سمول توك (SMALL TALK) و (C++).

وإذا قمنا بالمقارنة بين هذه اللغات بعضها ببعض نجد أن لغة لسب هي لغة طبيعية بينما البرولوج هي لغة منطقية تعتمد على التركيب والتفرع بينما تمتاز لغة سمول توك على المدي الواسع للجغرافيك وإستخدام الشبايبك، اما لغة البرولوج فقد طورها اليابانيون سنة ١٩٨١ كأداة لتنفيذ الجيل الخامس تتبع المنطق وتشرح العلاقات بين الاهداف .



شكل (٣-٨) محاولة إدخال المستخدم كطرف في تطوير النظام الخبير



شكل (٣-٩) النظام الخبير بعد تطويره بإدخال خبرة المستخدم

وتعتبر لغة سمول توك التي تعتمد على توصيف الأهداف من اللغات الجديدة المناسبة لبناء نظم الخبرة حيث يعرف الهدف (Object) في هذه اللغة بأنه عنصر أساسي يعرف بمكان في الذاكرة علي أنه جملة، ويمكن القول بأن لغة سمول توك لها العديد من الفوائد التي تشمل على القدرة على الاضافة الجديدة للأهداف بدون تعديل للشفرة الاصلية، وكذلك الربط الديناميكي، وتعتبر لغة ليسب ولغة برولوج من الدعام الأساسية والمنتشرة عالميا الآن لبناء هذه النظم، حيث ظهرت مشتقات لهما يمكنها التعامل مع اللغات الاخرى مثل لغة (ALS) (PROLOG)، وتعتبر لغة برولوج ذات الانتشار الكبير في أوروبا واليابان عن الولايات المتحدة الأمريكية لتنفيذ هذه النظم، بينما تعتبر لغة سمول توك هي لغة البرمجة بالاهداف والتي انتشرت في الولايات المتحدة الأمريكية. وأهم حزم البرامج الجاهزة هي (Small Talk 80). وبين شكل (٣-١٠) بعض حزم البرامج للبرمجة بالقائمة والمنطقية ومشتقاتهما.

<p>حزم برامج البرمجة بالقائمة (حاسبات شخصية)</p> <p>PC LISP Packages</p>	
Name of Package	Application PC
BYSCO LISP CLISP Cromenco LISP Expert LISP Golden Common LISP 286-Developer IQLISP LISP / 80 LISP / 88 muLISP UO - LISP Waltz LISP XLISP mu LISP 90 Lisp LIBRARY FOR C COMMON LISP CLOE 4.0	IBM IBM Z - 80 Macintosh / IBM IBM IBM IBM IBM IBM all IBM all all all all (UNIX 386/486) 386/486 Windows
<p>حزم برامج البرمجة المنطقية (حاسبات شخصية)</p> <p>PC Prolog Packages</p>	
Arity Prolog IF/Prolog LPA Micro-Prolog MPROLOG Micro-Prolog PC-Prolog PROLOG-1&2 PROLOG V Prolog-86 Turbo Prolog Rule Talk ALS Prolog(Prolog Compiler, GUI Tools, Object Oriented Prolog) Quintus Prolog/386. (Compatible with Quintus Prolog 3.1)	IBM IBM IBM all IBM IBM IBM IBM IBM IBM (MS Windows) Windows 3.1, OS/2, MAC. HP, IBM.

شكل (٣-١٠) بعض حزم البرامج للبرمجة بالقائمة

والبرمجة المنطقية ومشتقاتها.

(٣-٥) أدوات بناء النظم الخبيرة (Expert System Building Tools)

تعتبر الادوات المساعدة في بناء النظم الخبيرة من الاهمية ، حيث انها قد تطورت بشكل كبير في معامل البحوث المختلفة ثم جري تسويقها علي المستوي التجاري وبين شكل (٣-١١) التصنيف العام لهذه اللغات إعتمادا علي نوع اللغة المستخدمة في التطوير. ولقد كان مقياس مستوي النجاح للاداة في التطبيق هو إمكانية الاستخدام في اكثر من مجال ، ويمكن تقسيم مراحل التطور زمنيا كما يلي:

١ - المرحلة الاولى (الى ١٩٨٠)

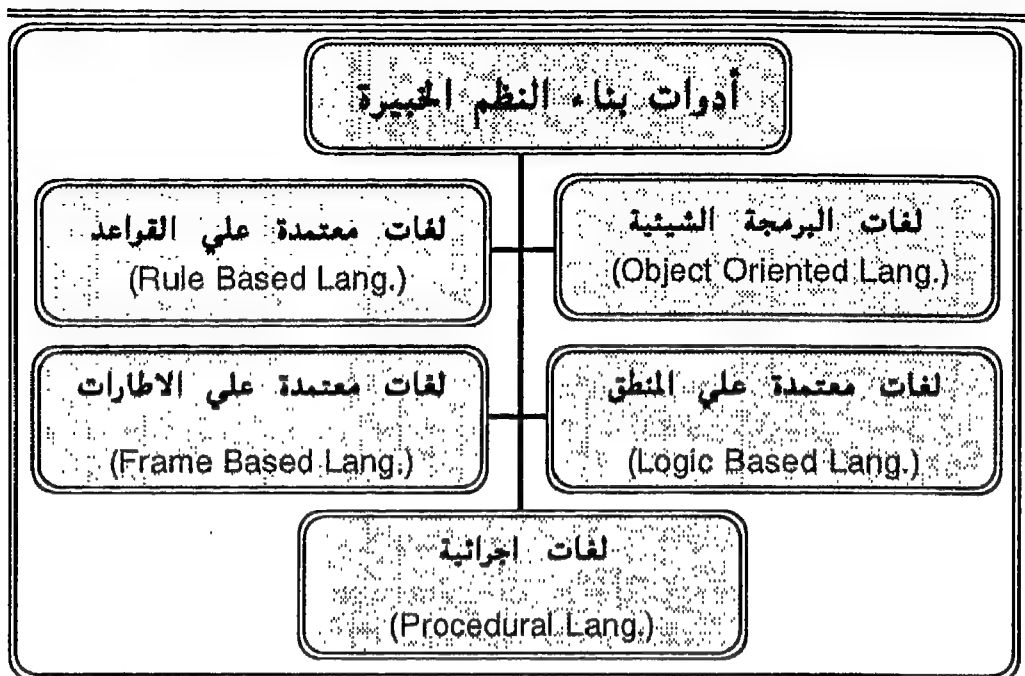
والتي إمتدت من الايام الاولى للذكاء الاصطناعي الي بداية الثمانينات من هذا القرن والتي تميزت بالبحث عن ادوات بناء ذات طبيعة عامة تفي باغراض اكثر من تطبيق مثل الاداة (GRS) ولكن ذلك الاتجاه سرعان ما تم إهماله وعدم الاعتماد عليه والاتجاه الي نظم معتمدة علي نوع المعرفة التي سوف تستخدم في بناء النظام الخبير ولقد أثبتت التجارب أن الاتجاه الجديد قد لاقى نجاحا كبيرا من الناحية الاقتصادية ولكن يشوبه الاعتماد على نوعية المعرفة في مجال التطبيق.

وتعتبر لغة السب من اقدم اللغات المستخدمة لبناء النظم الخبيرة والتي امكن إعتبارها كأساس مثل لغة التجميع (Assembler) حيث تم استخدامها بنجاح لبناء الكثير من الادوات بعد ذلك، وفي هذا الاتجاه تم تطوير ادوات مساعدة (Tool kits) وغلافات (Shells) وأدوات مساعدة ذات طابع خاص (Special Tool kits)، ويعتمد بناء النظام الخبير بوجه عام علي البنود الأساسية التالية :

١- الاستراتيجية المناسبة لتمثيل المعرفة.

٢- التفاصيل الدقيقة لعملية المواءمة.

وفي هذه الفترة الزمنية للتطور تم المحافظة على أن تكون هذه البنود منفصلة، مثال ذلك عند بناء النظام الخبير (MYCIN) تم الحصول منه على الأداة المساعدة (EMCIN) وهي عبارة عن النظام نفسه بعد تفرغ من محتوياته المعتمدة على نوعية المجال ماعدا التشكيل البنائي وكذلك محرك الاستدلال، وبذلك تم استخدام هذه الأداة في بناء نظم خبيرة أخرى. ولقد اعتمد هيكل أداة البناء (EMYCIN) على قواعد الانتاج والمعرفة والسياق الشجري (Context Tree) الذي يساعد على توجيه المستخدم، وكذلك نظام لقياس المصادقية معتمدا على نظرية التصديق، ولقد اعتمد محرك الاستدلال على ميكانيكية التسلسل الخلفي باستخدام إتجاه العمق

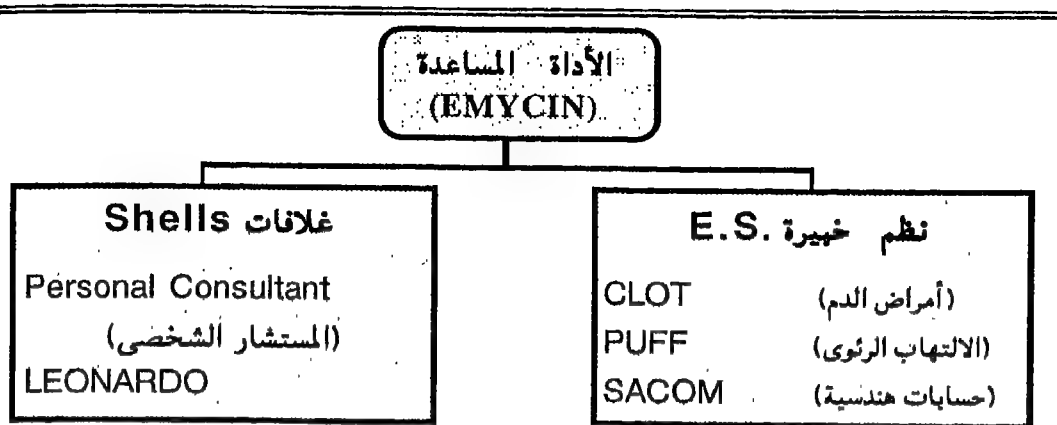


شكل (٣-١١) لغات البناء لادوات النظم الخبيرة

أولا. وتم استخدام الأداة المساعدة (EMYCIN) فى بناء بعض لنظم الخبيرة وكذلك بعض الغلافات كما هو مبين فى شكل (٣-١٢).

تعتبر أداة البناء (EMYCIN) مناسبة لبناء نظم ذات طبيعة موصفة مثل النظم الطبية مثلا، ولاتناسب النظم التى تتطلب التخطيط او التصميم او التوليف او المتعدد الأهداف، وبالمثل تم استخدام النظام الخبير (PROSPECTOR) فى بناء الغلافة (K.A.S) على أساس هيكله الذى احتوى على مجموعة من قواعد الانتاج فى شكل شبكة استدلالية تعمل من خلال شبكة دلالية تربط كيانات مختلفة فى المجال الجيولوجى، واعتمد محرك الاستدلال على التسلسل الخلفى وعدم مصداقية معتمدة على قاعدة (BAYES).

واستخدمت الغلافة (K.A.S) فى بناء النظام الخبير (HYDRO) الذى يساعد رجال الجيولوجيا على البحث عن وفرة المياه وكذلك استخدمت فى تطوير وبناء كثير من الغلافات مثل الغلافة (SAVOIR). وبالمثل استخدام النظام الخبير (CASNET) الذى استعمل فى تشخيص ومعالجة المياه الزرقاء (GLAUCOMA) فى تطوير الغلافة (EXPERT) مع استخدام لغة الفورتران فى تنفيذ القواعد، وتم الاستعانة بهذه الغلافة فى بناء النظم الخبيرة / (AI) (ELAS), (RHEUM0).



شكل (٣-١٢) استخدام الأداة فى تصميم نظم خبيرة وغلافات جديدة

ولقد أثبتت هذه النظم ان الاعتماد على القواعد فى تنفيذ النظم والغلافات قد صادفه النجاح بغض النظر عن بعض السلبيات البسيطة. ولقد تم استخدام اللغات المعتمدة على الاطارات (Frame-Based) فى بناء النظم الخبيرة حيث يعتبر النظام الخبير (INTERNIST) أول نظام خبير يجرى تمثيل المعارف على شكل مجموعة من الاطارات المتلاحقة بشكل وصفى، وتم بناء محرك الاستدلال على أساس عمل دورات تتمثل فى وضع الفروض ثم اختبار الحل.

ومن أهم مميزات هذه النظم هى أن المعارف التى يتم إدخالها فى الاطارات يمكن أن تكون مأخوذة من كتاب او مجلة مثلا، وبالمثل تم بناء النظم الخبيرة (PIP)، (IDT) وبناءاً على هياكل هذه النظم الخبيرة تم بناء الأدوات المساعدة مثل الغلافة (KMS)، والتى استخدمت بعد ذلك فى بناء النظام الخبير (STSTEM-D) والذى يتولى تشخيص مرض الدوخان (Dizziness).

ولقد قام رامزى (Ramsey) بعقد المقارنة بين الثلاث نظم الاتية للبناء وهى : النظم المعتمدة على تصنيف بايزان (Bayesian Classification)، النظم المعتمدة على القواعد، والنظم المعتمدة على الاطارات، وذلك من خلال استعراض ١٨ نظاما خبيرا وتوصل الى مايلى :

- ١- اعتماد كل نظام من هذه النظم على اساس نظرى يفى بمتطلبات إيجاد الحل المطلوب.
- ٢- صلاحية هذه النظم لبناء النظم يتطلب الاختيار لبديل واحد فقط من البدائل المطروحة.
- ٣- صلاحية هذه النظم لمجموعة عريضة من المجالات المختلفة.
- ٤- أن هذه النظم تفى بمتطلبات النظم الخبيرة ذات المعرفة الضحلة (Shallow Knowl- edge)، ولا تفى بمتطلبات النظم المقفلة (Closed Systems).

٢ - المرحلة الثانية (١٩٨٠ - ١٩٩٥)

والتي بدأت مع بداية الثمانينات من هذا القرن، حيث تم تطوير وظهور بعض اللغات ذات

الطبيعة الخاصة والتي تعرف بلغات المعرفة الهندسية - Knowledge Engineering Languages) حيث استخدمت لغة ليسب (LISP) لخلق وتطوير بيئات تتراوح بين الليسب والغلاقات اللازمة لبناء النظم الخبيرة، وبذلك تم تطوير عائلة (OPS) والتي بدأت بظهور (OPS4) ثم (OPS5) وهكذا. ولقد وقع اختيار مصممي النظم الخبيرة (RI) ، (XCON) على (OPS5) كأداة لبناء هذه النظم.

وتتميز هذه العائلة باستخدام التسلسل الأمامي وكذلك وجود أداة قوية للمواءمة (Matching Facilities) التي تساعد على حل التضارب تحت مسمى (MEA) ، (LEX). بعد ذلك قام المطورون بإنتاج (OPS83) التي تختلف في الأساس عن (OPS5) في أنها تتبع الطرق الاجرائية (Procedural) وتطوير لغة روزي (ROSIE) والتي تعتبر نظام معتمد على القواعد لتوصيف الخبرة المكتسبة (Rule Oriented System for Implementing Expertise) والتي استخدمت بنجاح في كثير من النظم الخبيرة العسكرية والاساسية مثل نظام اختيار الأهداف التكتيكية للطيران، ومكافحة التطرف الدولي، وفي مجال القانون. قتل لغة البرمجة المنطقية البرولوج الشق الثاني الاساسي لبناء النظم الخبيرة والغلاقات، حيث تعتبر هذه اللغة أكثر علوا من لغة ليسب، وتتميز بوجود التسلسل الخلفى وتمثيل العلاقات، والتي اتخذت كأساس لأنواع أخرى من البرولوج مثل:

(Turbo Prolog) ، (Quintus Prolog) ، (IF Prolog) ، (ALS Prolog) .

ولقد تم ادماج البرولوج والليسب العام (Common LISP) ، (POP-II) في لغة واحدة (WIMP) تستخدم النوافذ والفأرة والمشير. ولقد تم استخدام البرولوج كأداة لتطوير حاسبات الجيل الخامس وظهور لغة (KL1) ولغة المعارف (QUIX-OTE) في المشروع الياباني (١٩٨٠ - ١٩٩٢). وتمثل بيئة البرمجة الشيئية (Object Oriented) الشق الثالث في لغات البرمجة الاساسية للنظم الخبيرة، وذلك بتطوير لغة (Smalltalk)، وتم استخدام لغة الليسب لبناء نظم مثل (FLAVORS)، (LOOPS). كما تتميز هذه البيئة بأنها تحتوى على ارتباط الطريقة والبيانات (Procedures & Data) عن طريق الرسالة (Message) واعتمادها على صفة التوارث التي تستخدم الاطارات في التمثيل علاوة على سهولة المواءمة مع البيئة المحيطة بالنظام.

(٦-٣) الأدوات المساعدة (ToolKits)

تعتبر الأدوات المساعدة الآتية هي الأدوات التي تستخدم على المستوى التجارى:

١- الأداة ((KEE) Knowledge Engineering Environment): تعتبر أداة للبرمجة

فى البيئة الهندسية وهى أقدم الأدوات، ولقد تم تصميمها باستخدام لغة (UNITS) المعتمدة على هيكل بنائى للاطارات والمزودة بإمكانية الرسوم الجرافيكية، واعتماد الاستدلال على القواعد والتسلسل الامامى والخلفى، علاوة على استخدام البرمجة فى البيئة الشيئية وادخال الصور والنوافذ المتعددة للشرح وغير ذلك.

٢ - **الأداة (ART) Automated Reasoning Tool**: تعتبر أداة للاستدلال الآلى، وتعتمد على القواعد باستخدام نموذج السبورة، حيث يتم توصيف المعرفة المعلنة وتعريف العلاقات بينها. ومن أهم صفات هذه الأداة انها تستخدم الاستدلال المتغير النغمة والوتيرة (Non-Monotonic) وتعتبر كذلك سريعة العمل، حيث انه يمكنها تحديد المسار الأمثل.

٣ - **الأداة (Knowledge Groft)**: تم تطوير هذه الأداة باستخدام لغة (SRL) والتي تسمح بوجود أكثر من فرض واستخدم فى تصميمها لغة (CRL-Prolog) التي تعتمد على القوة الاشتقاقية للبرولوج والقوة التمثيلية للبرمجة الشيئية. تقوم هذه الأداة بتحديد المسار الأمثل للتوارث، واعتماد الاستدلال على قاعدة (CRL-OPS5) التي ادخلت قوة الاستدلال للأداة (OPS5) فى هيكل هذه الأداة، حيث انها تحتوى على فرعين للتشغيل، أحدها للبرولوج والآخر لـ (OPS5).

ويمكن القول بأن التطور للأدوات المساعدة فى التسعينات من هذا القرن قد ساعد على ظهور أدوات تعتمد على استخدام الاستدلال المعتمد على حل حالات سابقة او خبرة سابقة (CBR) (Case Based Reasoning) وذلك اعتمادا على وجود الحلول لمشكلات سابقة. ومن أهم هذه الأدوات مايلى :

١ - **أداة الاستدلال السهل (Eclipse 3.1 / The Easy Reasoner)**: حيث تعتبر أداة معتمدة على وجود الحلول لمشكلات سابقة، ومن أهم صفاتها : أ- تعتمد على طرق التمثيل العلاقى والشيئى. ب- تعدد فجوات الادخال (Slots) فى الطبقة الواحدة. ج- تحديد الأهداف آليا. د - الربط الديناميكى بين مكتبات الحالات السابقة والمخزونة بالبرنامج. هـ- التحكم والتجاوز الذكى من خلال النوافذ. و - يمكن استخدامها فى بيئة (Unix, MS-Windows, DOS)

٢ - **الأداة (ReCall)** تم تصميم هذه الأداة باستخدام النمط التركيبى المفتوح لكى يمكن اضافة حالات الاستدلال السابقة وحلولها، وتعتمد على البرمجة الشيئية فى طرق التوارث المتعدد، وقياس العلاقات، وتعدد طرق التحليل الآلى للحالات، والفهرسة. ويمكن ان تستخدم فى بيئات (MS-Windows, Unix, SUN Stations) ويمكن ان تستخدم للتحكم فى الاتصالات.

الفصل الرابع

المجالات التطبيقية
للنظم الخبيرة

Expert System's
Practical Applications

(٤-١) المجالات التطبيقية للنظم الخبيرة

نظرا لأهمية النظم الخبيرة فى المجالات التطبيقية والتي تم تنفيذها وتطويرها في المرحلة الحديثة الممتدة إلى الآن فلقد تم وضع معظم هذه النظم فى قاعدة بيانات مفهرسة طبقا للحروف الأبجدية تحتوي على ما يقرب من (١٢٤) من هذه النظم الخبيرة وموضحا بها مجال إستخدام كل منها والغرض الذى تم تصميمها من أجله.

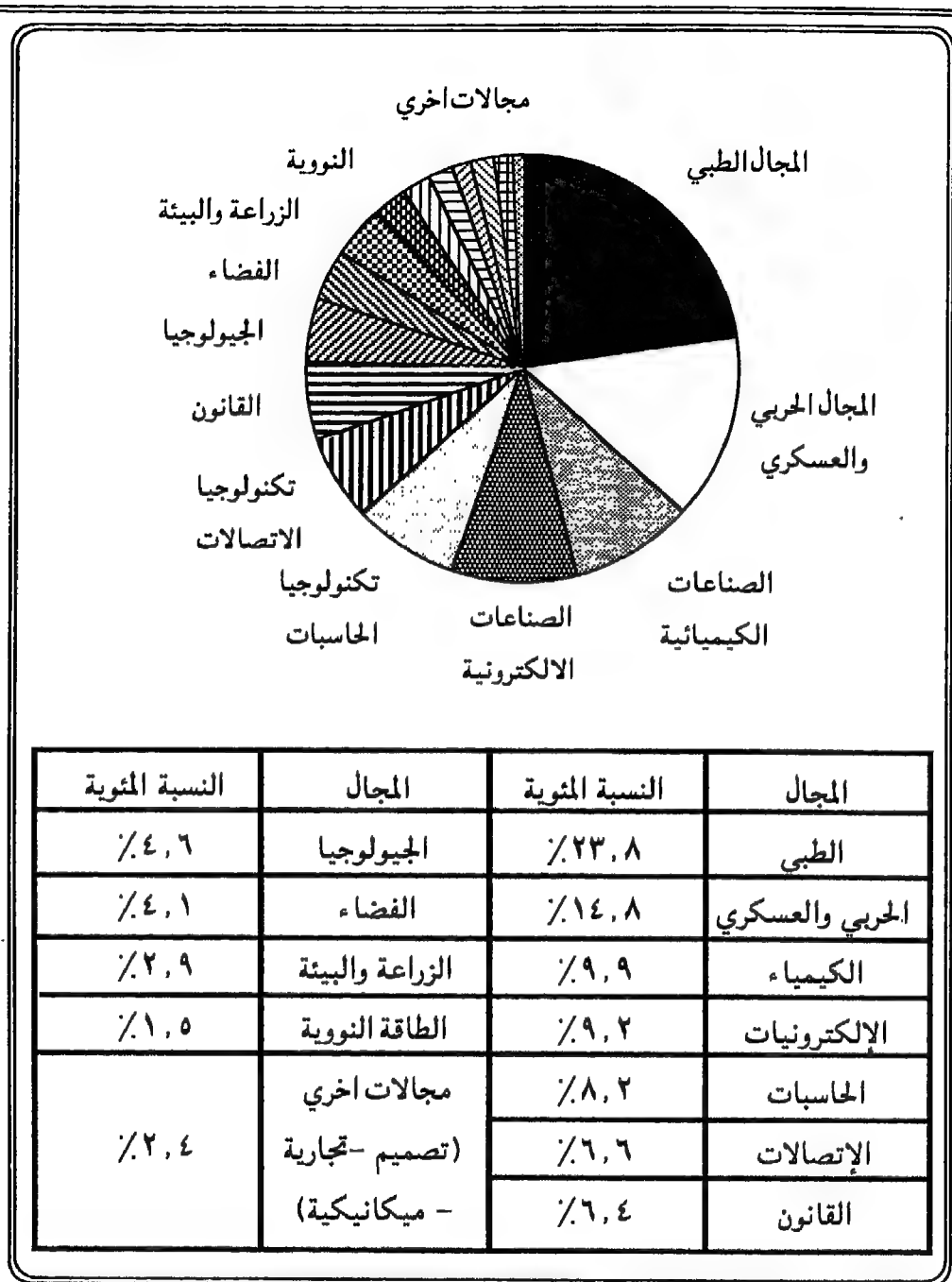
وبين شكل (٤-١) التمثيل الكروي والنسبة المئوية لمجال الاستخدام حيث يحتل المجال الطبي المركز الأول بنسبة ٢٣٪ وذلك في مجال التشخيص للأمراض مثل الأمراض الناتجة عن إختلال نسبة الحموضة فى الدم، وتحليل كفاءة الجهاز البولى والكلىتين، والأمراض الروماتزمية، وأمراض الشريان التاجي، والأمراض النفسية، وأمراض القلب العامة، واضرابات الصدر، وغير ذلك من الأمراض المختلفة.

ثم يليه المجال الحربي والعسكري بنسبة ١٤.٨٪ مثل التعرف على الاهداف الحيوية من الصور الجوية والرادارية، وتنظيم عمليات الاقلاع والهبوط للطيران العسكري، وتقدير الموقف بأرض المعركة، والسيطرة الآلية، والرصد، والتصنت، وغير ذلك.

ثم يلي ذلك مجالات التصنيع الكيميائي مثل بناء المركبات الكيميائية المعقدة، وتحليل المركبات الجزيئية، وتحديد التركيب البنائي بناءً على التحليل الطيفي، وحفظ واقتراح بناء المركبات الحيوية الشديدة التعقيد، والكيمياء الصناعية، وغير ذلك.

ثم يلي ذلك مجالات الإلكترونيات، والحاسبات، والإتصالات، والقانون، والجيولوجيا، والفضاء، والزراعة والبيئة، ثم مجالات الطاقة النووية، والتصميم، والتجارة، والميكانيكا وخصوصاً تكنولوجيا الانسان الآلى.

وتبين قاعدة البيانات (٤-٢) القائمة التطبيقية لبيانات النظم الخبيرة.



شكل (٤-١) التمثيل الكروي لمجالات الاستخدام التطبيقية والنسبة المئوية للنظم الخبيرة

(٤-٢) قائمة النظم الخبيرة (Expert Systems List)

النظام الخبير	المجال	الإستخدام
A		
ABEL	الطبي	تشخيص الأمراض الناتجة عن إختلال نسبة الحموضة فى الدم
ACES	الإتصالات	تحديد أماكن الأعطال فى الشبكات التليفونية والكوابل.
ADEPT	الحربى	تقدير وتقييم الموقف العسكرى الحربى بأرض المعركة.
AI-COAG	الطبي	تشخيص الأمراض الناتجة عن تجلط الدم.
AI-MM	الطبي	تحليل كفاءة الجهاز البولى والكليتين.
AIQ Market	التجارى	دراسة اقتصاديات السوق وتحويل البيانات الى نصائح تسويقية.
AI-RHEUM	الطبي	تشخيص الأمراض الروماتزمية والانسجة الضامة.
AIRID	الحربى	التعرف على نوع الطائرات الحربية من المعلومات البصرية.
AIRPLAN	الحربى	تنظيم عمليات الإقلاع والهبوط للطائرات.
AMF	الإتصالات	تحديد الأعطال وصيانة شبكة التليفونات TXE4A.
ANALYST	الحربى	تقدير وتقييم العمليات الحربية المباشرة بأرض المعركة.
ANGY	الطبي	تشخيص أمراض الشريان التاجى .
ASTA	الحربى	تحديد أنواع وطرقات الرادارات المعادية.
B		
BABY	الطبي	العناية بالأطفال حديثى الولادة بالعناية المركزة.
BATTLE	الحربى	إدارة النيران والمعاونة الجوية لمشاة البحرية الامريكية.
BDS	الإتصالات	تحديد الدوائر الإلكترونية المعطوبة بشبكات الإتصالات.
BLUE BOX	الطبي	تشخيص الأمراض النفسية وإقتراح العلاج .
C		
C-13	الكيمياء	تحديد التركيب الكيمائى لبعض المركبات العضوية المعزولة.

CADHELP	إلكترونيات	تصميم الدوائر الإلكترونية الرقمية باستخدام الحاسب.
CATE	الاتصالات	رفع كفاءة وتحسين أسلوب الاداء لشبكات التليفونات.
CLONER	الكيمياء	تخليق مركبات كيميائية حيوية جديدة.
CLOT	الطبي	تشخيص النزيف الناتج من تجلط الدم فى الاوعية.
CO-HLEX	إلكترونيات	نظام تصميم الدوائر المتوازية والمتكررة (اليابان ١٩٩٢).
CO-LODEX	إلكترونيات	تصميم الدوائر المنطقية عالية الكثافة (اليابان ١٩٩٢).
CONGEN	الكيمياء	تحديد التركيب البنائى بناءً على التحليل الطيفى.
CONPHYDE	الكيمياء	إختيار أنسب الخواص الكيميائية الفزيائية للمواد السائلة.
CRIB	الحاسبات	تحديد الأعطال بالحاسبات وإزالتها.
CRITTER	إلكترونيات	تصميم الدوائر الإلكترونية الرقمية عالية الكثافة (VLSI) .
CRYSLIS	الكيمياء	إستنتاج التركيب الثلاثى الأبعاد للبروتينات.
D		
DART	الحرى	إختيار أنسب نظم القيادة والسيطرة الآلية.
DELTA	الميكانيكا	تحديد الأعطال فى ماكينات الديزل وإقتراح طرق الاصلاح.
DENDRAL	الكيمياء	نظام خبرة فى الكيمياء العضوية ومطياف الكتلة.
DESIGNNET	الاتصالات	تصميم شبكات تراسل البيانات.
DIAGNOSE	الطبي	تشخيص أمراض القلب.
DIPMETER ADV	الجيولوجيا	تحليل البيانات الجيوفيزائية عند حفر الآبار.
DRILLING ADV	الجيولوجيا	تشخيص وتحليل مشاكل حفر الآبار البترولية .
DSCAS	القانون	الإجراءات القانونية لبنود العقود الانشائية.
E		
ECESIS	الفضاء	التحكم الأتوماتيكى لقيادة سفن الفضاء.
ELAS	الجيولوجيا	التحليل الجيولوجى لبيانات الآبار بهدف تقييم عملية الحفر.

EMERGE	الطبي	تحليل إضطرابات الصدر فى الإستقبال بالمستشفيات.
ENS	إلكترونيات	تصميم وإختيار الدوائر الإلكترونية الرقمية.
EPES	الحربى	الكشف وتحديد حالات الطوارئ بالمقاتلات ف١٦.
EURISKO	الإتصالات	إختيار وتحديد أنسب التصميمات للشبكات التليفونية.
EXAMINER	الطبي	نظام طبي لإختبار الاطباء وتحديد الكفاءة.
EXPERT NAVI	الحربى	إدارة النظام الملاهى للطائرات الامريكية التكتيكية.
F		
FAITH	الفضاء	تشخيص الأعطال فى سفن الفضاء.
FOREST	إلكترونيات	الكشف عن الأعطال بالاجهزة الإلكترونية بإستخدام الخبرة العملية
G		
GAMMA	الفيزياء	تفسير أطياف الأشعة النووية (جاما) وتحديد التركيب.
GDCC	الميكانيكا	تصميم الحركة الميكانيكية للانسان الآلى (اليابان ١٩٩٣).
GENESIS	الوراثة	تصميم نظم للتطبيقات فى الهندسة الوراثية.
GUIDON	الطبي	تصميم تعليمى لإختبار العلاج للمرضى بالبكتريا المعدية.
H		
HANNIBAL	الحربى	الرصد والتصنت للاتصالات المعادية.
HASP-SIAF	الحربى	تحديد أعماق المحيطات.
HEADMED	الطبي	تشخيص الأمراض النفسية وإقتراح العلاج.
HELIX-II	القانون	إقتراح الخطوات والبنود القانونية (اليابان ١٩٩٢).
HYDRO	الجيوولوجيا	تنظيم وتحليل متطلبات إنتاج وإستخدام المياه.
I		
I AND W	الحربى	تحليل الأوضاع السياسية والعسكرية والتنبؤ.
IMACS	الإنتاج	إقتراح الخطط التفصيلية لإنتاج النظم للحاسبات.

ISIS	الإنتاج	تنظيم وإقترح الإنتاج فى المصانع.
J		
JUDITH	قانون	تحليل القضايا وتحديد الإسنادات القانونية.
K		
KAPPA-P	حاسب	تصميم قواعد البيانات المختلفة (معارف).
KAT	الإتصالات	تحليل وتخطيط الشبكات التليفونية.
KNEECAP	الفضاء	إختيار خطة برامج مكوك الفضاء وتحديد الزمن اللازم.
KNOBS	حرى	التخطيط لمهام مراكز القيادة الجوية وتحديد الأولويات.
L		
LDS	قانون	تحديد المسؤولية القانونية فى الإنتاج الصناعى.
LES	الفضاء	التحكم فى إمداد الأكسجين السائل بمركز كيندى للفضاء.
LITHO	جيولوجيا	تحديد الخصائص البيئية للصخور.
LRS	قانون	تحديد التشريعات القانونية عند التعامل بالشيكات.
M		
MACSYMA	رياضيات	المعالجة الرياضية الرمزية.
MDX	طبى	تشخيص وإقترح علاج الأمراض الكبدية المزمنة.
MECHO	الفيزياء	دراسة النظم الميكانيكية مثل الشد والقصور الذاتى.
MECS/AI	الطبى	تشخيص أمراض الأوعية الدموية للقلب.
MEND. ZONE	الحاسبات	البرمجة الأوتوماتيكية للمعرفة (اليابان ١٩٩٢).
MEOMYCIN	الطبى	تشخيص أمراض الالتهاب السحائى.
MES	الحرى	تحديد الأعطال واصلاحها بالطائرات الحربية (تعليمى).
MGTP	الحاسبات	النمذجة واثبات النظريات المختلفة (اليابان ١٩٩٢).
MI	الطبى	تشخيص أمراض الذبحة الصدرية.

MIXER	الحاسبات	المساعدة فى إنتاج الشرائح السليكونية عالية التكامل.
MODIS	الطبى	تشخيص أمراض ضغط الدم الشريانى الزائد.
MOLGEN	الوراثة	إقتراح وبناء مركبات بالهندسة الوراثية.
MUD	الزراعة	إستخدام السوائل للحفر والمحافظة على سيولتها.
MYCIN	الطبى	تشخيص أمراض تلوث الدم بالبكتريا وإقتراح العلاج.
N		
NAVEX	الفضاء	المراقبة الرادارية لسفن الفضاء.
NET-ADVISOR	الإتصالات	تشخيص وكشف وتحديد الأعطال بالشبكات.
NEUREX	الطبى	تشخيص أمراض الجهاز العصبى وإقتراح العلاج.
NPPC	النووية	تحديد الظواهر الغير مألوفة بالمحطات النووية.
O		
OCSS	الكيمياء	تحليل التركيب الجزيئى بناءً على التركيب المقدم للمادة.
ONCOCIN	الطبى	تنظيم العلاج الكيميائى لمرضى السرطان.
P		
PALLADIO	إلكترونيات	التصميم الثلاثى الأبعاد للأجهزة الإلكترونية.
PATHFINDER	الطبى	تفسير وفحص أنسجة الغدد الليمفاوية.
PEACE	إلكترونيات	تصميم الدوائر الإلكترونية المتكاملة المنطقية الرقمية.
PEC	الطبى	تشخيص أمراض العيون والتوصية بالعلاج.
PLANT-CD	الزراعى	تشخيص وتحديد التلفيات ببنيات الذرة.
PLANT-DS	الزراعى	تشخيص وتحديد أمراض نبات فول الصويا.
POMME	زراعى	تقديم الارشادات اللازمة لإنتاج التفاح.
PROSPECTOR	الجيولوجيا	كشف المعادن والخامات وتحديد طرق البحث.
PTRANS	الحاسبات	إختبار وإنتاج نظم للحاسبات توائم طلب العملاء.

PUFF	الطبي	تشخيص أمراض الرئة وتحديد علاجها.
R		
REACTOR	النوية	تشخيص الأعطال والكشف عن الحوادث بالمفاعلات.
REDESIGN	إلكترونيات	تعديل التصميم للدوائر الإلكترونية بناءً على المتغيرات.
RODIN	الحاسب	نظم التوليف للدوائر عالية المستوى.
RTC	الحربي	التعرف على السفن الحربية من الصور الرادارية.
S		
SACON	الهندسى	تحديد الخواص الميكانيكية للمواد عند تعرضها للإجهاد.
SADD	إلكترونيات	تصميم الدوائر الرقمية بناءً على متطلبات الاستخدام.
SAL	القانون	تقييم الادعاءات عن التعريضات فى القضايا.
SARA	القانون	المساعدة فى تحليل القضايا المعروضة.
SCENARIO	الحربي	نمذجة التصرفات السياسية لدول العالم الثالث حيال الأزمات .
SECS	الكيمياء	حفظ وإقترح بناء المركبات الحيوية المعقدة.
SOPHIE	إلكترونيات	تقنيات كشف الأعطال الإلكترونية (تعليمى) .
SPAM	الحربي	التعرف على الأهداف الحيوية من الصور الجوية.
SPE	الطبي	تحديد أمراض الكبد والتفرقة بين السرطانات والتليف.
SPERIL-I	الهندسى	توقعات للخسائر بالمباني عند حدوث الزلازل.
SPERIL-II	الهندسى	تحديد وحساب درجة الأمان فى المنشآت عند حدوث الزلازل.
SPEX	الكيمياء	تخطيط التجارب العملية الكيميائية.
STEAMER	ميكانيكا	تشغيل المحطات ذات الرفع البخارى (تعليمى) .
SYNCHEM	الكيمياء	تحديد التركيب الجزيئى للمركبات العضوية.
SYNCHEM2	الكيمياء	التخليق المستقل للمركبات العضوية.
SYSTEMED	الطبي	تشخيص وتحديد أسباب الدوار.

T		
TALIB	التخطيط	يقوم بإقتراح الخطط عند تصميم الدوائر الإلكترونية.
TATR	الحربى	نظام إختيار الأهداف المعادية المناسبة للقوات الجوية.
TAXADVISOR	القانون	تحديد و إقتراح التسوية لمشاكل الضرائب.
TIMM-TYNER	الحاسبات	ضبط وتنظيم العمل والأداء فى حاسبات VAX
TQMSTONE	الكيمياء	تحليل وتفسير بيانات مقياس الكتلة.
V		
VM	الطبى	ملاحظة المرضى فى العناية المركزة بعد العمليات الجراحية.
X		
XCON	الحاسب	التصميم للحاسبات الإلكترونية بناءً على المتطلبات.
YES/MVS	الحاسب	المراقبة والتحكم فى نظام التشغيل MVS.

الفصل الخامس

حاسبات الجيل الخامس

Fifth Generation Computer Systems

(٥-٢) نظم الحاسبات (Computer Systems)

تعتبر نظم الحاسبات من أهم الأجزاء الحيوية فى علوم الحاسبات والتي تجسد العملية الحسابية وتحولها الى واقع، ومن المعروف أن نظام الحاسب يتكون من :

١ - المكونات المادية (Hardware) والتي تشمل المعالجات (Processors) والذاكرة (Memories) والاجزاء المكملة (Peripherals) وشبكات التوصيل وغير ذلك من المكونات المادية.

٢ - المكونات اللينة (Software) والتي تشمل نظم التشغيل والمفسرات وبروتوكولات الاتصال والبرامج التطبيقية.

من المعروف كذلك أن معظم الحاسبات التي تنتج اليوم يتم بناؤها علي أساس معمارية فون نيومان ذات الاغراض العامة للاستخدام والتي تعني انه طبقاً للبرنامج المستخدم يمكن ايجاد الحل للمشكلة او التطبيق المطروح والذي يتغير من تطبيق الى الآخر. ويشرح هذا الجزء من الكتاب المتطلبات اللازمة لتصميم نظم للحاسبات تفي بأغراض برامج الذكاء الاصطناعي بشكل عام، ومن المعروف ان برامج الذكاء الاصطناعي تشتمل على مايلي:

١- المعارف والتي تحتوي علي الكيانات والاشياء ومجالاتها الخاصة بالمشكلة قيد الحل وخواص هذه المجالات والعلاقات بينهما.

٢- آليات العمل علي هذه المعارف بعد وضعها في الشبكات الدلالية مثل التحليل والمواءمة للبصمات والتورث للصفات وماشابه ذلك.

ان معظم الحاسبات التي تنتج حالياً تعتمد على معمارية فون نيومان ذات الصبغة العامة والتي تعني انه يمكن برمجتها لتقوم بوضع الحل لتطبيقات مختلفة مثل التطبيقات التجارية والعلمية والصناعية وحتى برامج الذكاء الاصطناعي نفسها مع عدم مناسبتها لذلك. ونظراً لاعتماد هذه المعمارية علي المعالج والذاكرة حيث يقوم المعالج بالبحث عن الامر المراد تنفيذه من خلال الذاكرة ويجري توصيف هذا الامر بربطه بمكان معين معرف بالذاكرة مثال ذلك عملية الجمع فان المعالج يقوم بجمع الارقام الثنائية المخزونة بمكانين معرفين بالذاكرة ويقوم بخزن ناتج الجمع في مكان معرف ثالث. ومن هذا المنطلق يصبح تنفيذ برامج الذكاء الاصطناعي علي هذه الحاسبات فى احتياج الى:

١ - زمن تنفيذ كبير نسبياً وذلك لقيام المعالج بتخطيط المعارف المدخلة بالذاكرة ذات الطبيعة الخطية ومحاكات آليات العمل في شكل منطقي قابل للتنفيذ .

٢ - زيادة التكلفة حيث ان تنفيذ ادوات البرنامج مثل المترجمات ونظم التشغيل للقيام بالعمل علي تخطيط ومحاكاة وتنفيذ المعارف وآلياتها تصبح مرتفعة الثمن.

وفيمايلي اهم الاسباب التى تجعل معمارية فون نيومان غير مناسبة لتنفيذ برامج الذكاء الاصطناعى وهى:

١ - يعتبر عامل السرعة لهذه الحاسبات غير مجد وغير مناسب وذلك لان زيادة السرعة تعتمد علي زيادة سرعة إجراء العمليات الحسابية علي ارقام ذات نقطة عائمة واحدة بدقة غير مطلوبة عند تنفيذ برامج الذكاء الاصطناعي حيث تعتمد تراكيب البيانات علي القوائم والرسوم البيانية التي يجري تمثيلها رمزيا ورقميا والتي تحتاج الي المقارنة السريعة والموائمة للبصمات المختلفة .

٢ - تحتاج نظم الذكاء الاصطناعي عند تنفيذها الي المعالجة المتوازية حيث يجري تفتيت العمليات الي اجزاء يتم معالجتها علي التوازي ونظرا لقلّة وجود هذه التقنية في نظام فون نيومان فان تنفيذ هذه النظم يصبح عقيما ، ولقد جرت محاولات كثيرة لتطويع معمارية فون نيومان لتعمل في شكل متواز مثل استخدام تقنية الانابيب او تصنيع معالجات متوازية لكل منها ذاكرة صغيرة بحيث ان تكون كمية السيلكون اقل ممايكن ليبقي الثمن معقولا.

(٢-٥) نظم الحاسبات المعتمدة علي الذكاء الاصطناعي

(AI- Based Computer Systems)

تعتمد نظم هذه الحاسبات علي مكونات مادية مناسبة لطبيعة برامج الذكاء الاصطناعي والتي تعمل اساسا بشكل متواز وبذلك تصبح البرامج اللينة بسيطة للغاية اما المكونات المادية المناسبة هى التى تقوم بمعظم التنفيذ للبرامج ويمكن القول بأن هذه النظم سوف تفقد عمومية التشغيل للبرامج الاخرى الغير معتمدة على تقنيات الذكاء الاصطناعى. ومن هذه النظم :

١ - حاسبات لغة الليسب الرمزي ٣٦٠٠ (Symbolic 3600 Lisp Machine)

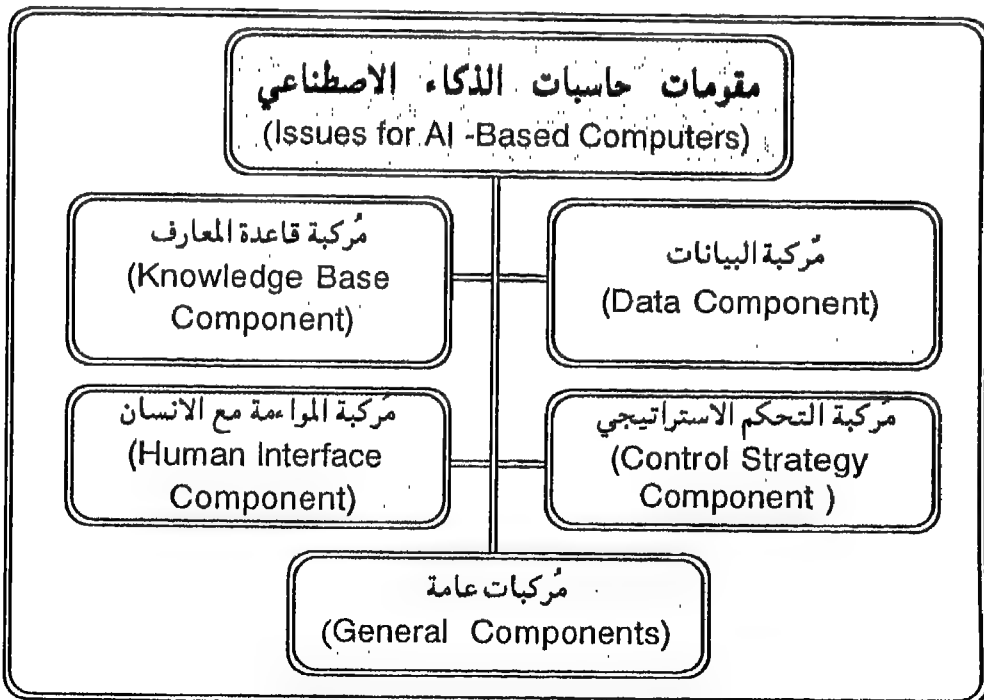
٢ - حاسبات الترابط (Connection Machine) والتي تعمل على أساس تمثيل المعلومات.

٣ - حاسبات الجيل الخامس والتي خرجت الى حيز التنفيذ فى الفترة من (٨١ - ١٩٩٢) من خلال مشروع الجيل الخامس اليابانى.

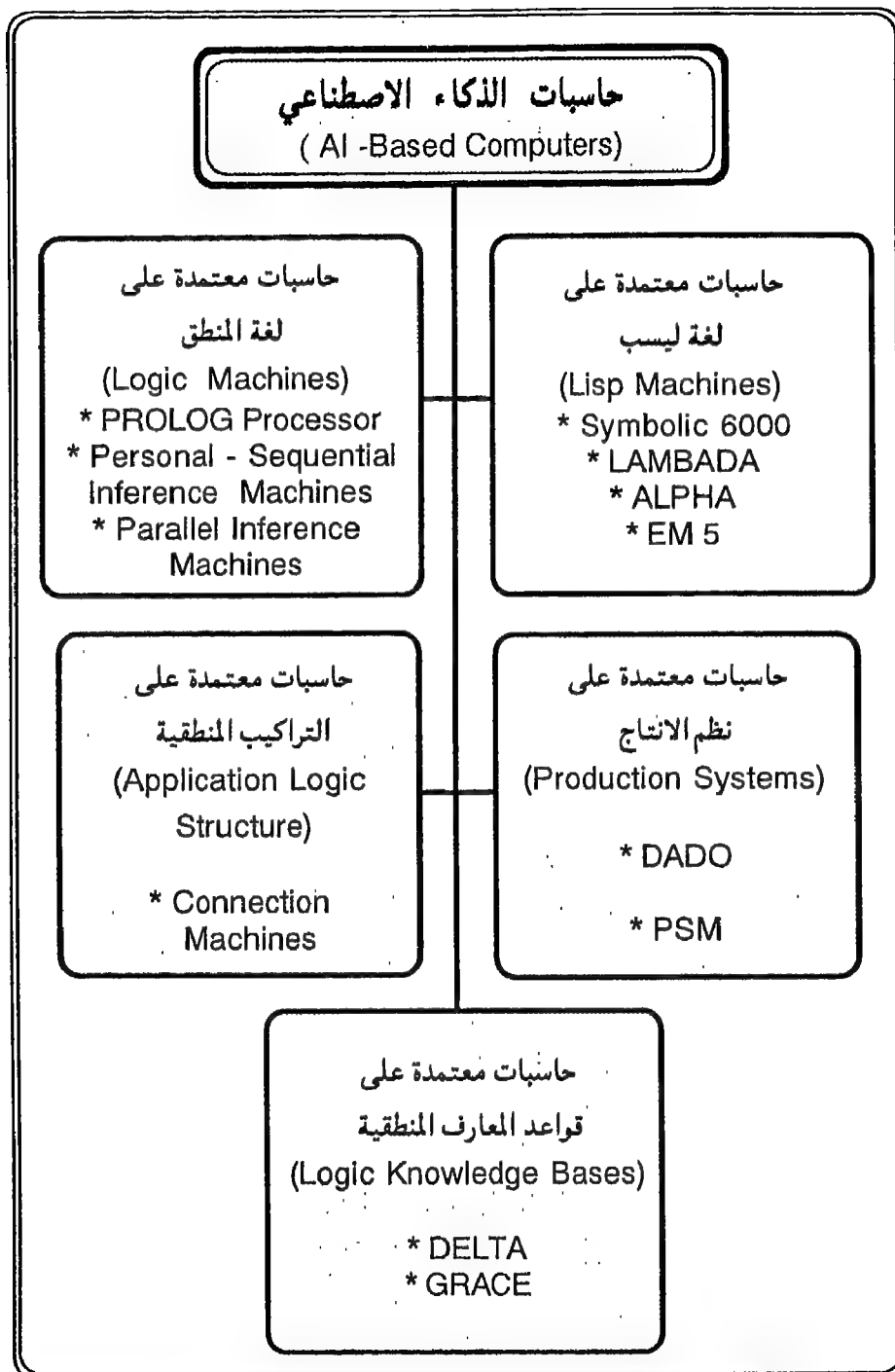
(١-٢-٥) مقومات الحاسبات المعتمدة على الذكاء الاصطناعي

(Issues for AI- Based Computer Systems)

- تشتمل نظم الحاسبات العادية على ثلاثة مُركبات هي البيانات والتحكم والمواءمة مع المستخدم، بينما تشتمل مقومات حاسبات الذكاء الاصطناعي كما فى شكل (١-٥) :
- ١ - مُركبة البيانات (Data Component) نظرا للطبيعة الاعلانية للبيانات واستخدام الطرق الرمزية والشبكات الدلالية والاطارات والمنطق الاحادي لتمثيل هذه البيانات فانه يلزم استخدام مكونات مادية مناسبة للقيام بدعم هذا التمثيل.
 - ٢ - مُركبة قاعدة المعارف (Knowledge Base Component) ونظرا لاعتماد آليات الذكاء الاصطناعي على عمليات مواءمة ومضاهات للبصمات والتوحيد والتحليل وتتبع خواص التورث فانه يلزم ان يصاحب ذلك جزء مصنع من المكونات المادية لدعم مثل هذه العمليات فى معالجة قواعد المعارف.
 - ٣ - مُركبة التحكم الاستراتيجى (Control Strategy Component) والتي غالبا ما تستخدم بعض تقنيات التحكم مثل طرق البحث الفراغى وتقنية نشر الشروط والقيود وتقنية اختصار المشكلة والتي غالبا ما تتم معالجتها علي التوازي وبذلك نصل الي نتيجة



شكل (١-٥) مقومات نظم حاسبات الذكاء الاصطناعي



شكل (٥-٢) حاسبات نظم الذكاء الاصطناعي

- أساسية لتكون عملية المعالجة على التوازي من خصائص المكونات المادية لهذه الحاسبات.
- ٤ - مُركبة المواءمة مع الانسان (Human Interface Component) والتي تشتمل على نظم ادارة الذاكرة ونظم ادارة التخزين وطرق تنظيم وجدولة عمل المعالج وبين شكل (٥-٢) بعض نظم حاسبات الذكاء الاصطناعي والتي تنقسم الى :
- ١ - حاسبات معتمدة على لغة اللسب.
 - ٢ - حاسبات معتمدة على لغة المنطق.
 - ٣ - حاسبات معتمدة على نظم الانتاج.
 - ٤ - حاسبات معتمدة على التراكيب المنطقية.
 - ٥ - حاسبات معتمدة على قواعد المعرفة المنطقية.

(٣-٥) حاسبات الجيل الخامس (المشروع اليابانى)

(Fifth Generation Computer Systems)

يعتبر المشروع اليابانى لإنتاج الجيل الخامس للحاسبات فى الفترة من (١٩٨١-١٩٩٢) أهم الانجازات التى ساهمت فى دفع عجلة التقدم فى مجال الذكاء الإصطناعى، ولقد بدأ المشروع عام ١٩٨١ بخطة قومية طموحة لإنتاج حاسبات الجيل الخامس والقادرة على محاكاة الإنسان فى المعالجة الرمزية (Symbolic Processing) والتي دعت الصحافة الالمانية إلى القول بان المشروع اليابانى يمثل الخطة الطموحة لإنتاج الآلات الحاسبة المفكرة (Thinking Machines) ولقد اشترك فى هذا المشروع ١٨٤ باحثاً يابانياً تم اختيارهم من ١٩ مؤسسة يابانية للأبحاث العلمية، كما ساهم ٧٥ باحثاً من ١٢ دولة أخرى لمدة قصيرة ومتوسطة تتراوح بين أقل من شهر وأكثر من سنة، وتم نشر أكثر من ٢٠٠٠ نشرة علمية وفى يونيو عام ١٩٩٢ تم الاعلان عن نتائج المشروع فى مؤتمر حاسبات المستقبل (FGCS) الذى عقد باليابان والذى تم فيه تقييم المشروع عالياً.

(١-٣-٥) الخطة الأساسية للمشروع

تم الاعلان فى عام ١٩٨١ عن الخطة الأساسية للمشروع والتي تتلخص فى بناء جيل من الحاسبات يشتمل على نظم وبرمجيات ولغات تكون قادرة على مايلى :

- ١ - الترجمة الآلية الأوتوماتيكية القورية (Automatic Translation) بين العديد من اللغات والتي سوف تؤدي الى زيادة التفاهم بين الشعوب .

- ٢ - بناء نظم المعالجة للمعلومات المعتمدة على المعارف (Knowledge In-formation Processing Systems)
 - ٣ - زيادة إمكانيات وأساليب الذكاء الاصطناعي وتسهيل استخدامها من قبل الإنسان كى تساعده على حل مشاكل التحدث (Speech) والصوت (Voice) والرسوم الجرافيكية (Graphics) والصور (Images) والوثائق (Documents) وذلك باستخدام اللغات الطبيعية الحية (Natural Languages) المستخدمة يومياً فى البلدان المختلفة، والقدرة على التعلم (Learning) والاستدلال (Reasoning) .
 - ٤ - تسهيل وسائل إنتاج البرمجيات الملائمة للمعالجة والبرمجة الآلية الأوتوماتيكية (Automatic Programming) وتطويع البرامج الحالية للاستخدام المستقبلى المناسب للنظم الجديدة وخلق البيئة المناسبة لازدهار وتطور صناعة البرمجيات.
 - ٥ - خفض التكاليف الإقتصادية وتحسين الأداء عند تطوير التقنيات الآتية :
 - أ - تقنيات بناء الدوائر الإلكترونية ذات الحجم التكاملى الكبير (VLSI) .
 - ب - تقنيات المعالجة المتوازية (Parallel Processing) وتقنيات الإتصالات (Communication) .
 - ج - تقنيات صناعة البرمجيات (Software Industry) .
 - د - تقنيات وسائل الذكاء الاصطناعى لبناء حاسبات خفيفة الوزن وعالية السرعة وذات قدرة تخزينية عالية وذلك لتجابه التطور فى النواحي المدنية والعسكرية.
- يمكن القول بأن أهداف المشروع هى الوصول إلى حاسبات تكون قادرة على الاجابة الأوتوماتيكية على الاسئلة التى توجه إليها (Question Answering) والقادرة على الترجمة الأوتوماتيكية (Automatic Translation) والفهم للحديث والصور وذلك بتطوير نظم معالجة المعارف (Knowledge Information Processing) وبناء حاسبات مستقبلية تعتمد على معماريات (Architecture) مختلفة عن معمارية (فون - نيومان) والتى يمكن أن تعتمد على تقنيات مثل تقنيات تدفق البيانات (Data-flow) والتى تسمح بتنفيذ أساليب البرمجة الدالية المنطقية لباكوس، حيث يصبح توصيف المشكلة والحل لها متماثلين (Backus's functional Programming) . ولقد تم تطوير اتجاهات البحث فى الأربعة مجالات الآتية :
- ١ - مجال اللغات الطبيعية وخصوصاً اللغة اليابانية وتشمل التركيب والتفهم ونماذج المحادثات.
 - ٢ - مجال تمثيل المعلومات والاستدلال والعلاقة بين قواعد البيانات والمنطق ومشاكل التغلب على الأحجام الكبيرة للمعارف وطبيعتها المعقدة ومحاولة تطوير المنطق للتغلب على ذلك.

٣ - مجال هندسة البرمجيات والتي تشمل وضع المواصفات للبرامج وأساليب التحقق ومشاكل التحويل بين البرامج المختلفة.

٤ - مجال معمارية الحاسبات والبحث عن معمارية جديدة لتطوير حاسبات تدفق البيانات وحاسبات تعمل على أسس (جبرية - علاقية) والخوارزميات اللازمة للمعالجة المتوازية.

(٤-٥) نتائج مشروع الجيل الخامس

(١-٤-٥) المشروعات اليابانية (١٩٨١-١٩٩٢)

بلغ عدد المشروعات التي تم الاعلان عنها ٢٠ مشروعاً، والمبينة في شكل (٣-٥) والتي غطت كثيراً من مشروع الخطة المستهدفة من قبل عام (١٩٨١) والتي شملت نظاماً خبيرة تعمل في مجال التشخيص والتحكم اعتماداً على النماذج المدخلة، وكذلك استحداث نظاماً للمساعدة في إنتاج وتخليق الدوائر والمكونات الإلكترونية عالية التكامل، ومحاكاة المنطق المتوازي، والدخول في مجال الميكروبيولوجي لتحليل البروتينات، حيث يتطلب ذلك استخدام المنطق المتوازي والحاسبات المتوازية المتطورة في التحليل.

كذلك الإثبات الأوتوماتيكي للنظريات الرمزية ولغات تمثيل المعارف ومجال اللغات الطبيعية حيث تم تخليق النصوص اليابانية في القانون والتحليل التركيبي للكلمات والجمل، وإيجاد العلاقات المنطقية بين الجمل اليابانية المختلفة، وتصميم الروبوتات، وصناديق الإرسال الأوتوماتيكية. وقد أثبتت هذه المشروعات نجاح المشروع، وتم استخدام لغة (Kernel Lan) 1) والمعتمدة على البرولوج كلغة أساسية.

(٢-٤-٥) حاسبات الإستدلال المتوازية

تم الاعلان عن حاسبات الجيل الخامس والتي تعمل باستخدام المعالجات المتوازية وذلك للوصول إلى سرعات تصل إلى ١٠٠ مليون وحدة إستدلال في الثانية وذلك في حالة استخدام (٥١٢) معالجا و (١٠٠٠) مليون وحدة إستدلال في الثانية في حالة استخدام ١٠٠٠ معالج على التوازي كما هو موضح في شكل (٤-٥).

مشروعات الجيل الخامس الياباني (١٩٨١-١٩٩٢)

(2) Experimental adaptive model -based diagnostic system.

(٢) نظام إنضباطي للتشخيص
إعتماداً على النموذج

(1) A Diagnostic and control Expert System based on plant model.

(١) نظام خبرة للتشخيص والتحكم
معتمداً على النموذج للمنشأة.

(4) Co-HLEX Experimental parallel hierarchical recursive layout system.

(٤) نظام لإختيار مكان المكونات
الإلكترونية عند التصميم.

(3) Case - based circuit design support system.

(٣) المساعدة في تصميم الدوائر
الإلكترونية بطرق الإستدلال.

(6) RODIN High level synthesis system

(٦) نظام تخليق للدوائر الإلكترونية
عالي المستوى

(5) Parallel cell placement experimental system.

(٥) نظام لإختيار مكان الدوائر
المتكاملة عند التصميم.

(8) Parallel LSI router

(٨) اختيار المسارات بين
نهايات الدوائر المتكاملة

(7) Co-LODEX

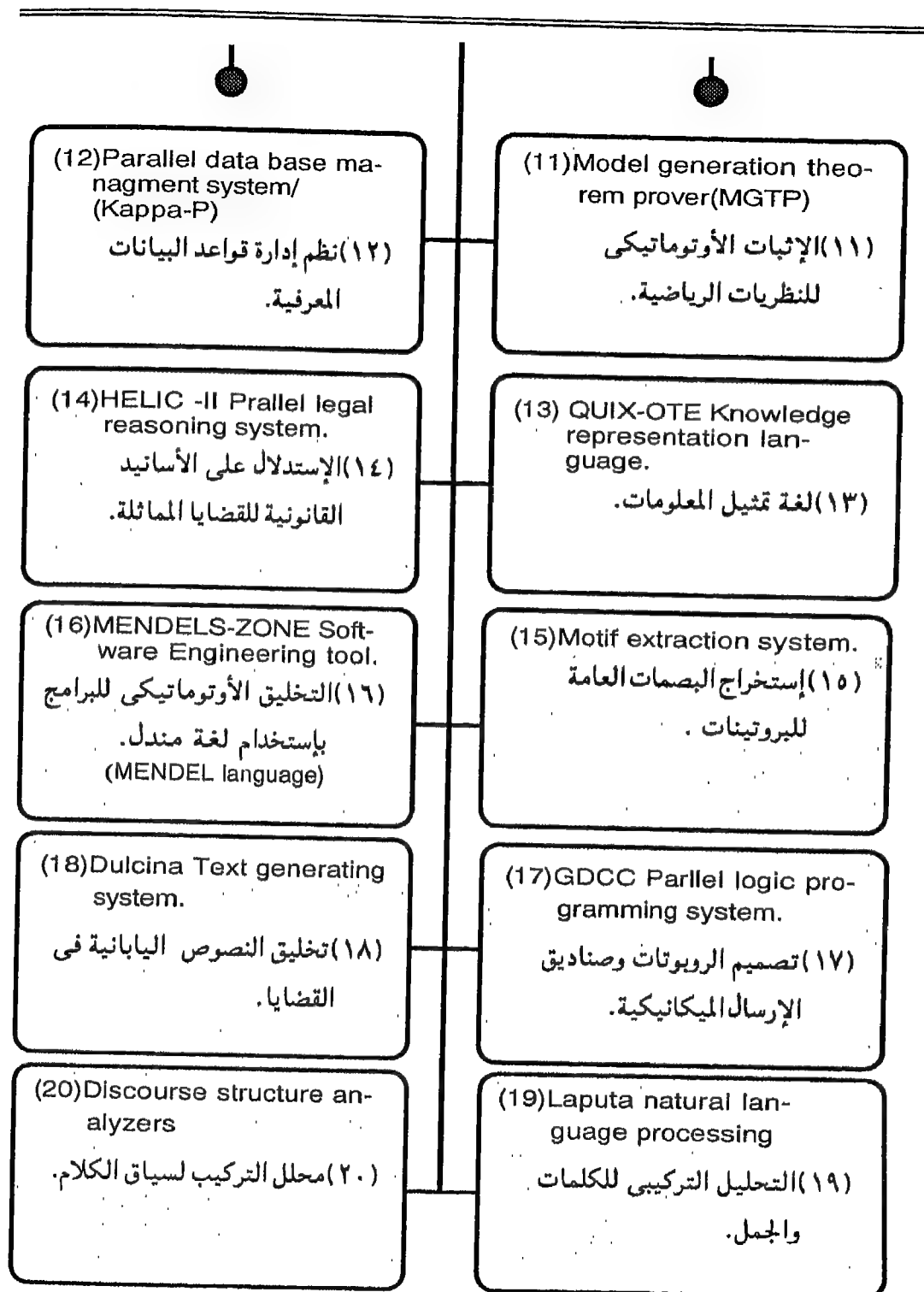
(٧) نظام خبرة لإختيار الجودة للدوائر
الإلكترونية (VLSI)

(10) Protein sequence analysis program.

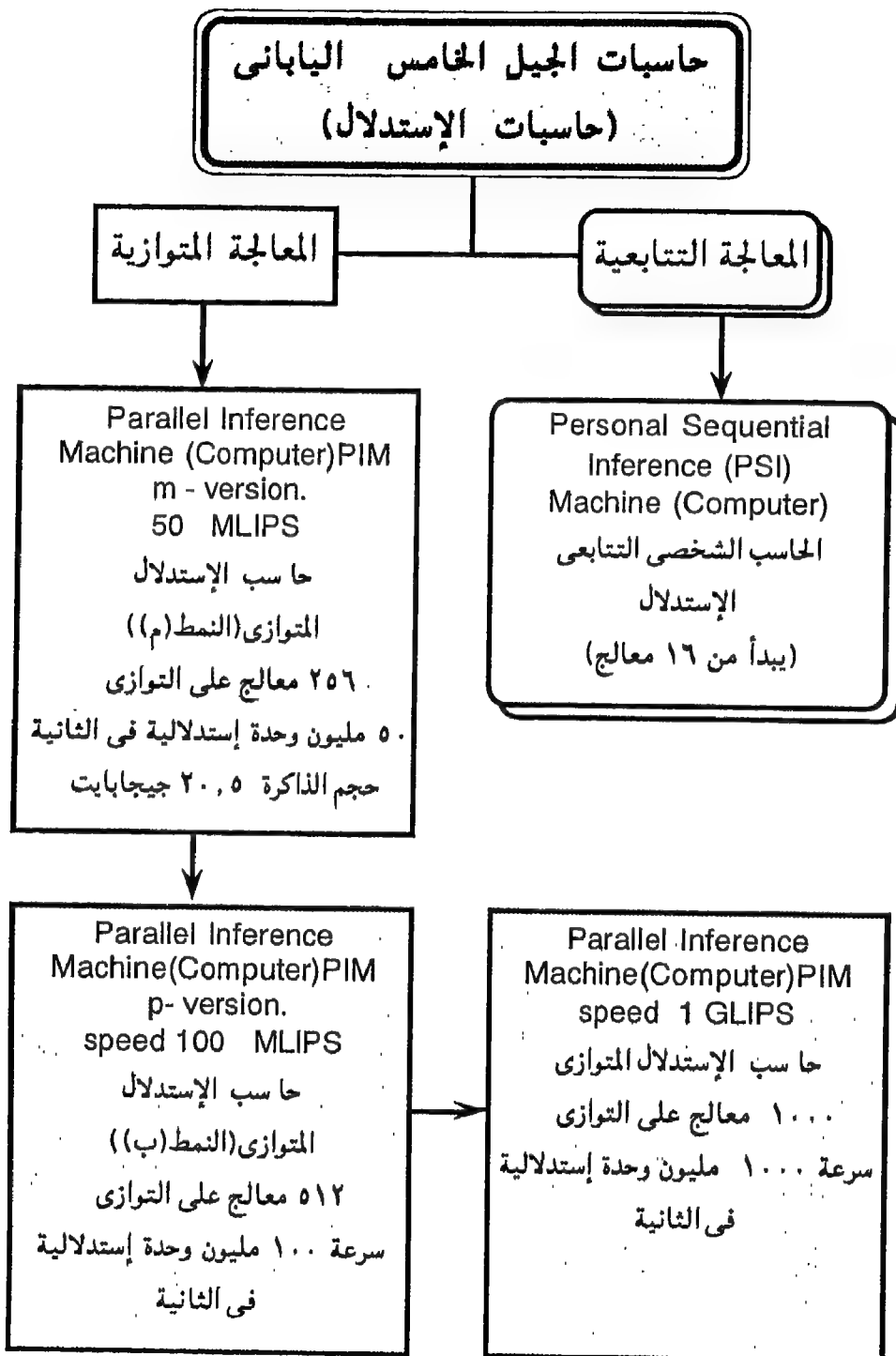
(١٠) نظام لتحليل التسلسل المتعدد
للبروتينات

(9) Parallel logic simulator

(٩) محاكاة المنطق المتوازي



شكل (٣-٥) مشروعات الجيل الخامس الياباني



شكل (٤-٥) حاسبات الإستدلال للجيل الخامس الياباني

الجزء الثانى

المعالجة الرمزية Symbolic Processing

الفصل السادس

الأسس الرياضية للمعالجة الرمزية Mathematical Basics for Symbolic Processing

(٦-١) الأسس الرياضية للمعالجة الرمزية

تعتمد دراسة الذكاء الإصطناعي على كثير من فروع علم الرياضيات مثل نظرية المنطق (Theory of Logic) وفي هذا الباب يتم القاء الضوء على بعض الأسس الرياضية التي تستخدم في المعالجة الرمزية والرقمية مثل الدوال (Functions) والرسوم البيانية (Graphs) والرسوم البيانية الموجهة (Digraphs) والشبكات الدلالية (Semantic Network).

(٦-١-١) المعرفة - الوصف والعلاقات (Knowledge - Description, and Relations)
بيان العلاقة اللغوية بين شيئين (x, y) والذي يمكن إعتبارهما رموزاً أو أرقاماً أو أفعالاً أو فروعاً أو بديهيات يمكن كتابة الآتى :
يبلغ التوصيف (R) للشيء (x) القيمة (y).

attribute R of object x has value y

$y=f(x, R)$ حيث يمكن كتابة (y) بدلالة (x), (R) كما يلي :

وإذا كان نطاق (R, x) يمثل مجموعة محدودة فإنه يمكن كتابة هذه العلاقة اللغوية بالشكل والترتيب كالاتى :

والتي تأخذ الشكل النمطي الآتى : (الوصف ، الشيء ، القيمة)

(Attribute, Object, Value)

(A, O, V) والتي يمكن إختصارها بالشكل الآتى :

والتي يمكن قراءتها كما يلي : التوصيف (A) للشيء (O) هو (V) .

ويمكن إستخراج العلاقة (Relation) بين (x, y) وذلك بكتابة

(x is related to y by R)

والذى يعنى أن (x) ترتبط بعلاقة مع (y) بالوصف (R)

ويكون التمثيل لهذه العلاقة بالرسم (Graphically) المقابل :

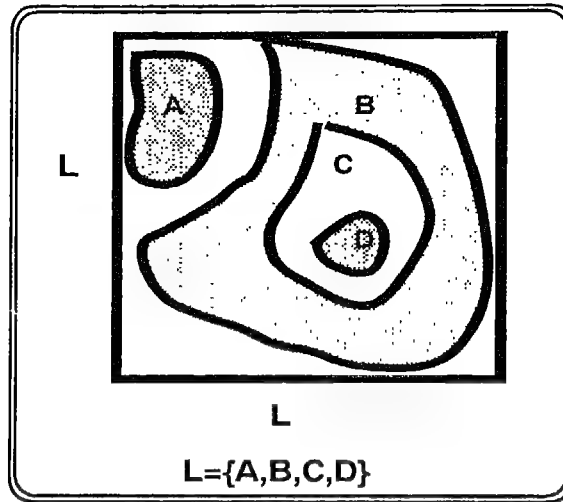


(٦-١-٢) تصنيف العلاقات

يمكن تصنيف العلاقات كما يلي:

١ - العلاقات الثنائية (Binary Relations)

إذا فرضنا أن هناك مجموعتان (A و B) متصلتان بعلاقة ما فإنه يمكن تعريف هذه العلاقة على أنها علاقة ثنائية إذا أمكن تكوين مجموعة جزئية (Subset) قيمتها حاصل الضرب الاحداثى للمجموعتين $(A \times B)$



شكل (١-٦) الترميز المكاني للمجموعات

٢- العلاقات الرمزية المكانية الإجرائية

(Positional & Procedural Symbolic Relations)

إن العلاقات التي تحتوى على كميات رمزية لها أهمية كبرى فى دراسة الذكاء الإصطناعى.

ولتفهم معنى العلاقات الرمزية المكانية والاجرائية نورد المثال الآتى :

إذا فرضنا أن هناك صورة شكل (١-٦) والتي تحتوى على مناطق منعزلة ومناطق متداخلة

كما هو واضح هي الشكل، وإذا تم ترميز (Coding) المناطق بالحروف (A,B,C,D) فانه

يمكن تعريف أية منطقة بأنها تمثل أحد العناصر المكانية المرمزة من المجموعة الآتية :

$$L=\{A,B,C,D\}$$

وتكون العلاقة بين هذه الرموز هي المجموعة الجزئية الناتجة من حاصل الضرب (LXL) والتي

يمكن تمثيلها بالرسم البياني الموجه (Digraph) كما فى شكل (٢-٦).

(٣-١-٦) العلاقات والصفات

إن فرض مفهوم التوصيف للخواص (Attributes) من الأهمية فى دراسة الذكاء

الإصطناعى، حيث يرتبط هذا المفهوم بإنشاء النماذج (Models) ويمكن أن تندرج أشياء

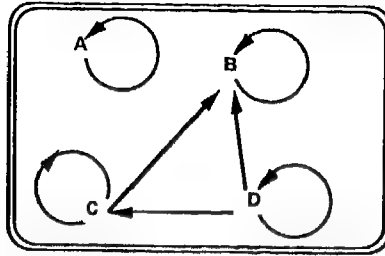
مختلفة مثل اللون والسن والعمر والعمل والتطبيقات تحت مسمى توصيف الخواص. مثال ذلك إذا

أردنا أن نمثل الحقيقة التالية :

" أن الحاسب يحتوى على دوائر متكاملة (computer has chips) "

فباستخدام التعريف السابق فان العلاقة (has) سوف تقوم بين الشئائى

(computer,chips) لتأخذ الشكل الآتى :



شكل (٦-٢) الرسوم الموجهة التي تمثل العلاقة المكانية المرمزة

has (computer, chips)

من هنا نرى ان العلاقة التي تمثل خاصية ما مثل (has, is_a, could_be) سوف يتم وضعها في النماذج بسهولة من خلال الشبكات الدلالية حيث تنقسم خواص العلاقة (R) والناشئة بين شيئين واللذين يمكن وصفهما بالمجموعتين (A, B) الى ثلاثة أقسام :

١- خاصية الانعكاس (Reflective) :

أي أن R تكون انعكاسية إذا كانت جميع قيم a جزء من A و (a,a) جزء من R

٢- خاصية التماثل (Symmetric) :

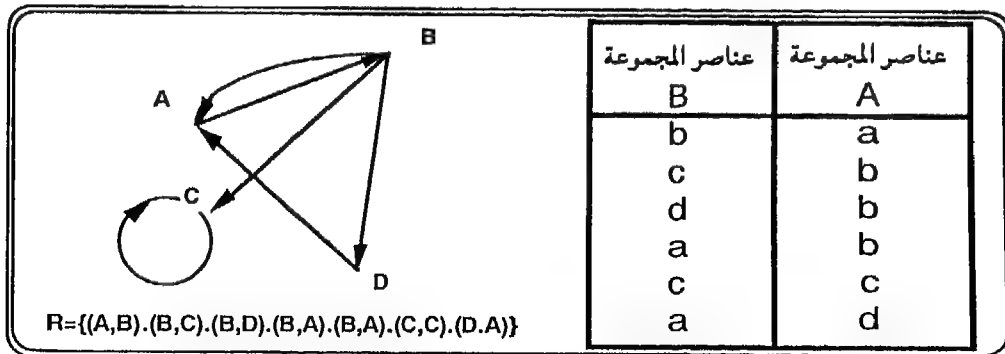
أي أن R تكون تماثلية إذا كانت جميع قيم a,b جزء من R وكذلك (b,a) جزء من R

٣- خاصية الامتداد (Transitive) :

أي أن R تكون امتدادية إذا كانت جميع قيم (a,b) جزء من R و (b,c) و (c,a) جزء من R.

(٦-١-٤) العلاقات غير الثنائية (ذات الرتبة اكبر من ٢)

ان العلاقة بين اكثر من شيئين (ثلاثة او اربعة او اكثر) والتي تعرف بالعلاقات غير الثنائية والتي تضع كثيرا من الشروط فانه يمكن وضع الحل لها عن طريق إيجاد جداول تتناول العلاقات الثنائية بين كل شيئين على حدة ويبين شكل (٦-٣) العلاقة البيانية الموجهة بين أربعة اشياء (A,B,C,D). كما يبين الجدول هذه العلاقة.



شكل (٦-٣) العلاقة البيانية الموجهة بين أربعة كيانات.

(٦-٢) الشبكات الدلالية (Semantic Nets)

يعتبر إستخدام الرسوم البيانية الموجهة (Digraphs) لتحديد العلاقات الرقمية والرمزية بين مجموعة من الأشياء عملاً أساسياً عند تمثيل المعارف، وأن توخى الدقة فى التمثيل يؤدي الى الوصول الى الحل بنجاح. تعرف الشبكة الدلالية بانها مجموعة من الرسوم البيانية الموجهة بالأسهم والقادرة على تمثيل العلاقات والخواص بين الأشياء والتي يمكن أن تشمل أوضاع معينة او نظريات وضعية، ويمكن اعتبار الشبكات الدلالية علي انها أداة للمراقبة البصرية (Visualization Tool) والتي تستخدم كثيراً فى عملية البدء (Get Started) فى تمثيل المعرفة. إن التركيب الطبيعي للشبكات الدلالية سوف يعكس توارث الصفات (Inheritance) لسهولة تتبع هذه الصفات من خلال الشبكة حيث يجرى تمثيل الشيء بعقدة (Nodes) او دائرة وتمثل العلاقة بقوس معرف بالكتابة لهذه الصفة (Labeled Arcs). لتوضيح تمثيل الأشياء والعلاقات باستخدام الشبكة الدلالية يبين شكل (٦-٤) منضدة عليها ثلاث مكعبات مختلفة الأحجام وبينهما علاقات كثيرة كما يوضح الشكل ايضاً الشبكة الدلالية لتمثيل المعرفة لهذه المكعبات. ويوضح شكل (٦-٥) الشبكة الدلالية الممثلة لجزء من سيارة، وبالنظر الى الشبكتين يمكن وضع الملاحظات الآتية :

١- تحتوى الشبكة الدلالية على كثير من العلاقات والتي تشمل :

is_a (is_smaller, is_larger....).
has_attribute (has_part,.....)
used_for
adjacent_to
shape_is

٢- تحتوى الشبكة الدلالية على علاقات عكسية مثل :

right_of, left_of
is_below, is_above

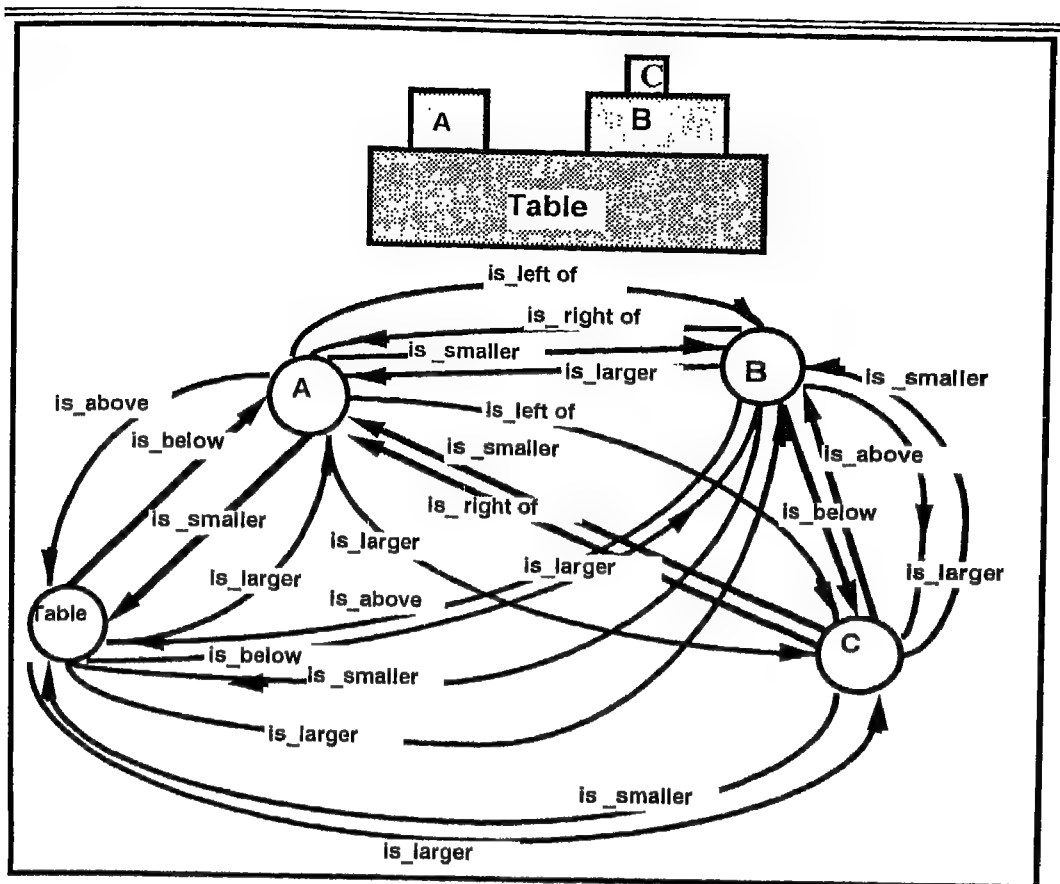
وتعتبر هذه العلاقات متكررة ولذلك يكتفى بواحدة فقط مما يؤدي الى تبسيط الشبكة.

٣- بعد تبسيط الشبكة يمكن تحويلها بسهولة الى خوارزميات مثل إستخدام المنطق الوصفى

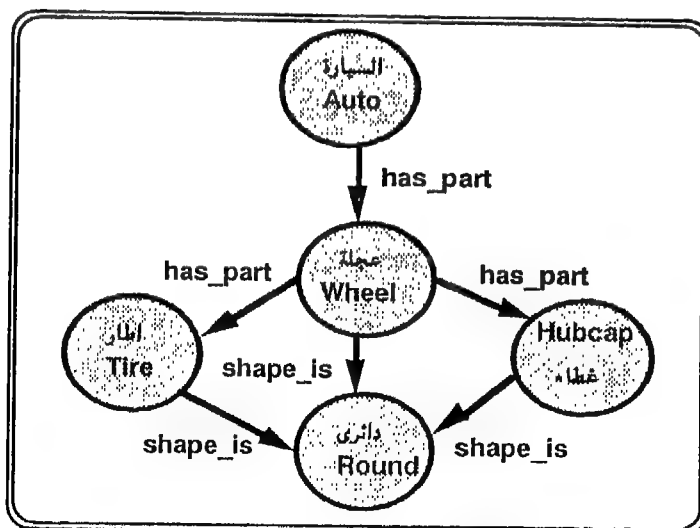
on_top_of(B, table) فى الشكل الآتى :

has_part(Wheel, Tire)

٤- يمكن إستخدام الشبكة فى دراسة أى تغير فى العلاقات بالنظام مما يؤدي الى ظهور شبكات أخرى تساعد فى إيجاد الشكل العام الجديد كما هو معروف مثلاً فى لعبة الشطرنج.



شكل (٤-٦) الشبكة الدلالية لتمثيل المعارف بين المكعبات والمنضدة



شكل (٥-٦) الشبكة الدلالية لتمثيل جزء من سيارة

الفصل السابع

النمذجة الحاسوبية

وتمثيل المعارف

Computational Modelling

and

Knowledge Representation

(٧-١) النمذجة الحسابية (Computational Modelling)

تعتبر النمذجة الحسابية وتمثيل المعارف من الناحية العلمية والهندسية هي الحصول علي نظام واضح المعالم والخواص يجرى فيه تمثيل الخواص الأصلية للمعرفة في المستوى الفرضي وإختيارها من هذا المستوى وتحديدتها الى اشياء مطلوب دراستها ثم تمثيلها واستخدام هذا النموذج الحسابي لإيجاد الحل للمشكلة المعروضة، ومن أجل ذلك فإنه عند محاولة وضع نموذج حسابي يناسب الإستخدام في الذكاء الإصطناعي يمكن تعريف الثلاثة مراحل الآتية للنمذجة الحسابية والتمثيل المعرفي والمبينة في شكل (٧-١) كمايلي:

١ - المرحلة الأولى : مرحلة تعريف المعرفة (Knowledge Definition)

وتشمل المستوى الفرضي (M) والذي يجرى فيه تمثيل الأشياء والمشكلة المطلوب دراستها بدقة والفرقة بينها وبين الاشياء الغير مطلوب دراستها.

٢ - المرحلة الثانية: مرحلة تمثيل المعرفة (Knowledge Representation)

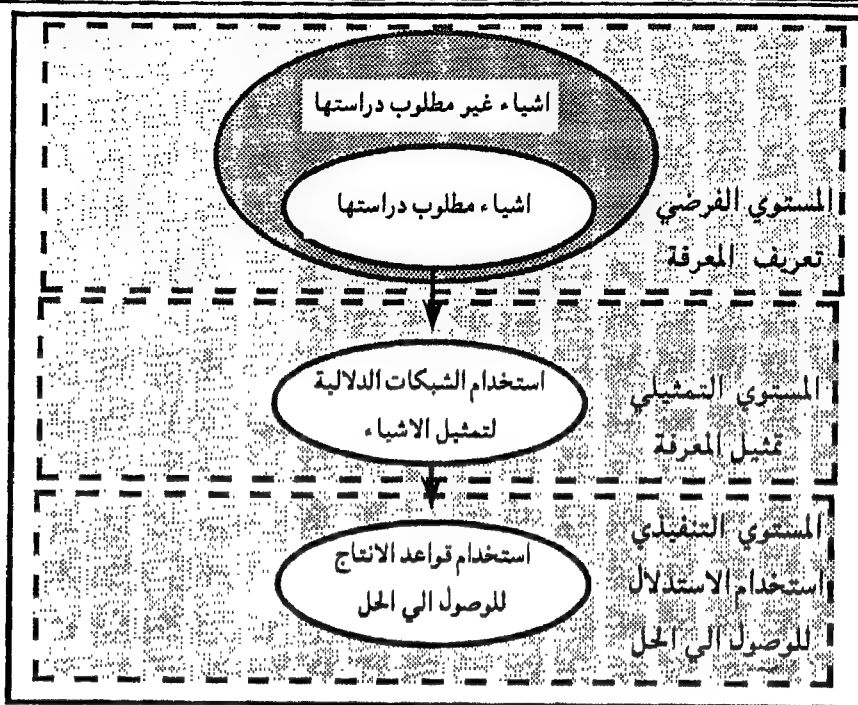
وذلك في المستوى التمثيلي (R) والذي يرتبط بالمستوى الأول بالعلاقة (S) وفي هذا المستوى تستخدم الشبكات الدلالية لتمثيل الأشياء وصفاتها.

٣ - المرحلة الثالثة : مرحلة الاختصار والعمل على إيجاد الحل باستخدام

الإستدلال وذلك في المستوى التنفيذي أو إستخدام قواعد الانتاج مثلا للوصول الى هذا الحل، ويطلق عليها مرحلة الإستدلال المبني على النماذج (Model Based Reasoning) (MBR).

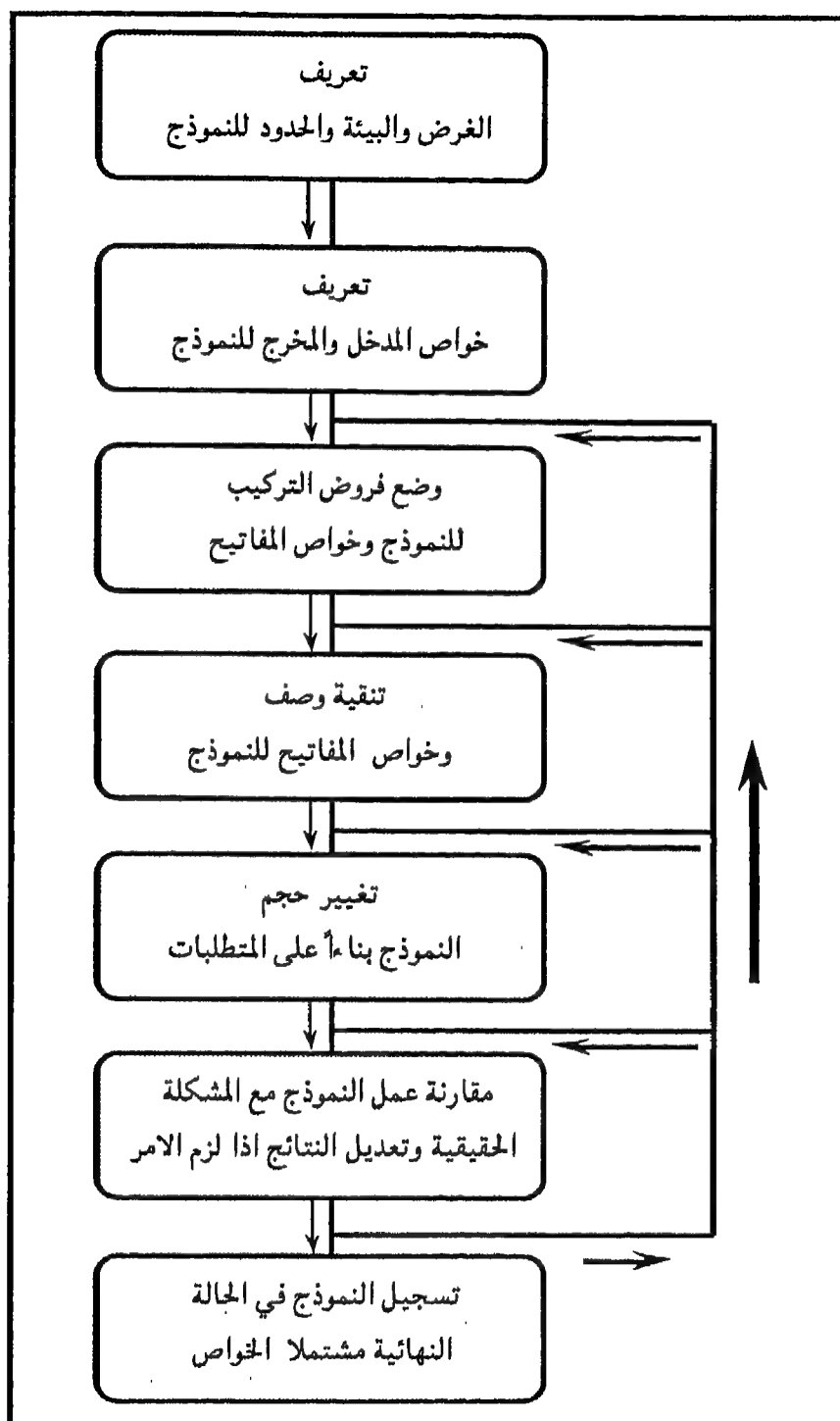
من الملاحظ أن إنشاء وتطوير النماذج ربما لا يصل الى التمثيل الكامل للنظام الأساسي ويعتبر الذكاء الإصطناعي من وجهة النظر الهندسية على أنه بناء وتمثيل نماذج مثالية أقرب الى الحقيقة والواقع وذلك في نطاق معين، ومن ثم تصميم وتطوير اكثر من طريقة للإستدلال للوصول الى الحل المناسب، ويعتبر الوصول الى حل بمثابة اصابة الهدف (Goal)، ويمكن تعريف تمثيل المعرفة بانه تحديد العناصر الأساسية لمشكلة ما في نطاق معين وإنشاء نموذج حسابي يتم إستخراج النتيجة منه باستخدام محركات الإستدلال (Inference engines) او الموحدات (Unifiers) التي تقوم بالعمليات الحسابية. ومن هنا نرى أن تمثيل المعرفة هو حجر الزاوية في عمل النماذج، ولذلك يراعى عند تصميم النماذج أن تكون هناك اجابة على الاسئلة الآتية :

- ١ - هل النموذج قابل للامتداد والتعديل بالزيادة او النقصان ليلائم دراسة نظاما أخرى في نفس المجال او في مجالات اخرى ؟
- ٢ - ماهي حدود التطبيق للنموذج ؟



شكل (٧-١) المراحل الأساسية للنمذجة الحسابية وتمثيل المعارف

- ٣ - هل يقوم النموذج بالتمثيل الحقيقي للمعرفة ومدى كفاءة التمثيل ؟
 - ٤ - هل يمكن للمستخدم التحكم فى متغيرات النموذج بسهولة ؟
 - (٧-١-١) مراحل إنشاء وتطوير النماذج (Levels of Model Development)
 - يبين شكل (٧-٢) الخطوات الواجب اتباعها عند إنشاء النماذج وتشمل الخطوات الآتية:
 - ١ - تعريف الغرض والبيئة والحدود للنموذج وذلك من خلال دراسة المشكلة الحقيقية وبيئتها واستخلاص الغرض والحدود التى سوف يتم فيها تطبيق النموذج.
 - ٢ - تعريف خواص المدخل والمخرج للنموذج .
 - ٣ - وضع فروض التركيب للنموذج وتحديد المفاتيح وخواصها .
 - ٤ - تنقية وصف وخواص المفاتيح للنموذج .
 - ٥ - تغيير حجم النموذج بناءً على المتطلبات .
 - ٦ - مقارنة عمل النموذج مع المشكلة الحقيقية، وأجراء التعديل إذا لزم الأمر مع الرجوع الى أى من الخطوات السابقة (٣، ٤، ٥) الى أن يصبح النموذج مطابقا تماما للمشكلة قيد الدراسة .
 - ٧ - تسجيل النموذج مشتملا على الخواص فى الحالة النهائية .
- ولتوسيع دائرة تمثيل المعارف عند إنشاء النماذج يمكن الإستعانة بالانواع المختلفة من



شكل (٧-٢) الخطوات الواجب إتباعها عند إنشاء النماذج الحسابية

التصنيف الآتى للمعارف وهى :

- ١- المعرفة المختصرة (Intensional Knowledge) هى المعرفة التى تحمل ملخص لمعنى او الإستخدام لفرض من الفروض.
- ٢- المعرفة عن مكونات الفرض (Extensional Knowledge) وهى المعرفة عن عناصر الفروض.
- ٣- المعرفة الذاتية (Meta- Knowledge) وهى المعرفة عن المعرفة.
- ٤- معرفة النطاق او المجال (Domain -Knowledge) وهى المعرفة التى توصف نطاق الموضوع قيد البحث وتشمل الفروض والحقائق والعلاقات بين هذه الحقائق والاجراءات اللازمة للوصول الى الحل.
- ٥- المعرفة الممثلة هرميا (Heurachical Representation) وتشمل تمثيل عناصر المعرفة بشكل هرمى.

(٧-٢) خواص التمثيل لنظم الإستدلال المبني على النماذج

(MBR Characteristics)

من أهم الضروريات التى يجب اتباعها عند إنشاء النماذج أن يكون التمثيل واضحا وممرزا ترميزا يمكن تتبعه بسهولة، وأن يضم القواعد والأساسيات والخواص التى تبرز خصائص الذكاء المطلوب، وأن يحتوى النموذج على :

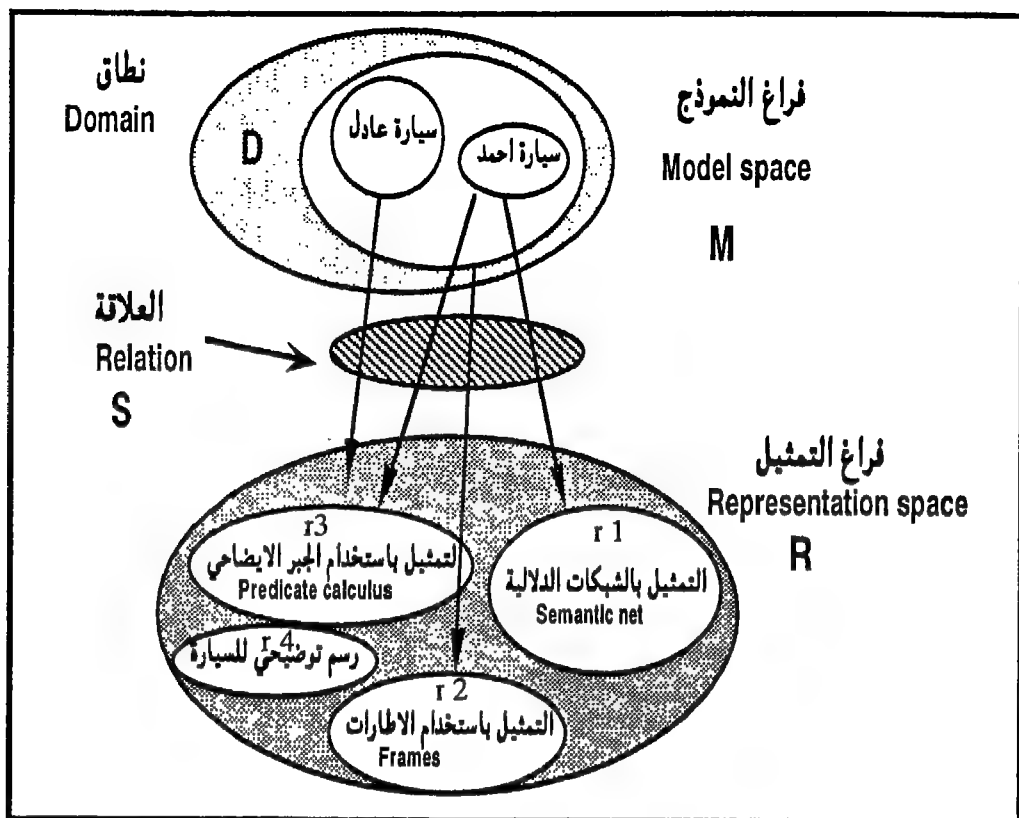
- ١ - الفروض مثل تقسيم النظام الى اجزاء ونظم مصغرة.
 - ٢ - الاشياء وعناصرها ويجب أن توضع فى ابسط صورها واصغرها (ذرة العنصر).
 - ٣ - الخواص والوصف والقيم للاشياء مثل اللون والوزن والعمر وما الى ذلك والقيم المختلفة.
 - ٤ - القيود والشروط والتى تشمل العلاقات بين الأشياء.
- الشروط الواجب توافرها بالنماذج الحاسوبية :

- ١- القدرة على التمثيل (Applicability) : أن يمثل النموذج تماما خواص النظام المطلوب والبيئة المناسبة لقدرات الذكاء الاصطناعى المطلوب.
- ٢ - التكامل (الوصفى) (Completeness) : إذا امكن للنموذج أن يحتوى جميع العلاقات وأن يصف بدقة جميع التحويلات بين عناصره فإنه يجرى تعريف هذا النموذج بالنظام المغلق (Closed System) ومن هنا تاتى صعوبة التمثيل او إنشاء النماذج.

- ٣ - المرونة وقابلية الامتداد (Flexibility & Extendibility) : من الأهمية أن يكون التمثيل او النموذج مرناً لكي يمكن أن يشمل التطبيق للتوقعات والتغيرات التي تحدث بعد تطبيق النموذج فى النظام الاصلى وأن يكون قابلاً للزيادة التى تقابل الزيادة فى النطاق او النظام الاصلى، مثال ذلك أن يكون النظام الطبى للتشخيص لمرض معين قادراً على أن يمتد ويشمل التشخيص لأمراض مماثلة.
- ٤ - التطابق (Consistency) : أن يكون النموذج قادراً على عدم خلق حلول متعارضة وأن تكون النتائج متطابقة فى جميع الحالات المتماثلة.
- ٥ - قابلية التتبع (Tractability) : أن يكون النموذج قادراً على التتبع الإجرائى.
- ٦ - قابلية التكامل مع البيانات الهيئية (Integrability with Environmental Data) أن يسمح النموذج بإدخال الكميات والبيانات التى يمكن قياسها من البيئة.
- ٧- قابلية التطبيق (Practicality) : أن يكون قابلاً للتطبيق من الناحية العملية.

(٣-٧) القواعد الهندسية للنمذجة

- تعتمد النمذجة فى الذكاء الاصطناعى على أسس النمذجة الهندسية (براون ١٩٨١) والتى أمكن تطويرها لتشمل مجال الذكاء الاصطناعى كما فى شكل (٣-٧) كالاتى :
- ١- فراغ النموذج M (Model Space): يعرف فراغ النموذج على أنه الفراغ الذى يشمل العناصر التى تحمل تلخيص لجميع الأشياء اللازمة لدراسة المشكلة فى النطاق المعين مثلاً إذا تعلقت المشكلة بدراسة الموانع فانه يلزم الاستعانة بقوانين الفيزياء والميكانيكا وكذلك المعادلات التفاضلية علاوة على أصل المشكلة فى هذا الفراغ، او كما هو واضح فى الشكل عند المقارنة بين سبابة احمد وسيارة عادل فإن هذا الفراغ سوف يحتوى على السيارتين.
 - ٢- فراغ التمثيل R (Representation Space): وهو الفراغ الذى يحتوى على التشكيل الرمزي مثل التمثيل بالشبكات الدلالية $(r1)$ والتمثيل بالإطارات $(r2)$ والتمثيل بالجبر الخبرى او الايضاحى $(r3)$ وكذلك الرسوم الايضاحية للسيارة $(r4)$.
 - ٣- خطة التمثيل S (Representation Scheme): وهو العلاقة بين فراغ النموذج (M) وفراغ التمثيل (R) والتى تمثل القوة الوصفية للخطة، ويعتمد التمثيل (R) على قيم (S) ، فاذا كانت (S) تعبر عن دالة فإن التمثيل يكون وحيداً (unique)،



شكل (٧-٣) العلاقات الهندسية (الفراغية) لأسس النمذجة الحاسوبية
 ويكون التمثيل فقيرا في التعبير او بعيدا عن الغرض إذا كانت $(1/S)$ لها قيمة
 واحدة. ويعتبر التمثيل $(r1), (r2)$ متطابقا في التمثيل الفراغي (R) إذا كان هناك عنصر
 واحد على الاقل (m) في تمثيل فراغ النموذج (M) مرتبطا بالعلاقة :

$$S1(m) = r1$$

$$S2(m) = r2$$

(٧-٤) التمثيل باستخدام الإطارات

(Representation using frames)

إن تمثيل المعرفة عن طريق إستخدام إطارات متتابعة في أزمنة متتابعة بحيث يشمل الإطار
 على حالة الأشياء والعلاقات والحقائق في الزمن اللحظي المعين، وبذلك يتحول النموذج الى عدد
 من الإطارات المتعاقبة كما هو مبين بالشكل (٧-٤) حيث يبين الإطار الأول أن التصادم سوف يتم
 بين الكرتين كما هو مبين في الإطار الثاني وبذلك تتحول المشكلة الى عدد من الاطارات
 المتلاحقة، ومن هنا تنشأ مشكلة الإطارات في الذكاء الاصطناعي حيث أنه من الممكن عدم حدوث

التصادم فى النظام الحقيقى نتيجة عوامل أخرى لم يتم تمثيلها مثل الاحتكاك بين الكرتين والسطح، مما يؤدى الى عدم صحة التوقع.

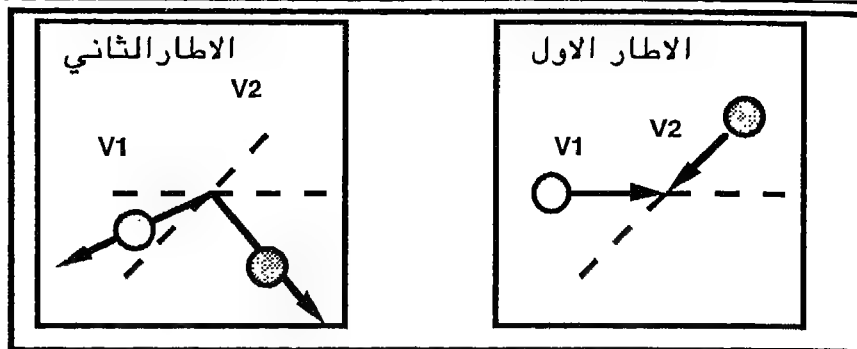
ومن هنا جاء تعريف مشكلة الإطارات والتي تنص على ما يلى : " عند إستخدام التمثيل بالإطارات فإنه يجب تمثيل الأشياء والقواعد الثابتة قبل المتحركة فى كل الإطارات " وتظهر مشكلة الإطارات بشكل واضح عندما تكون هناك الحاجة الى إستخدام الإستدلال الإضطرابى (Temporal Reasoning) والتخطيط (Planning) واستخدام الحدث او الشعور الادراكى (Common-Sense). ويمكن وضع الملاحظات الآتية على الإطارات الغير كاملة التوصيف كما فى المثال السابق:

- ١ - تكون التوقعات غير صحيحة لإعتمادها على معلومات ناقصة.
 - ٢ - ميل الإنسان الى التسرع فى الإستدلال بناءً على الملاحظة النظرية من أول وهلة.
 - ٣ - تداخل المعلومات الرمزية والرقمية مثل قوانين الحركة فى الإستدلال.
 - ٤ - تغير التوقع بعد لحظة صغيرة من بدأ التحرك.
 - ٥ - تكون التوقعات أقرب الى التوقع الصحيح اذا كانت الإطارات فى فترات صغيرة من الزمن.
- الأوضاع (Situations)**

من أهم الأساسيات عند الاستعانة بالإطارات فى التمثيل اتباع الأوضاع والتي تعرف بأنها التصور اللحظى للنظام فى لحظة معينة، والذي يشابه الصورة الواحدة الثابتة فى الافلام السينمائية حيث تظهر الصورة من تلاحق هذه الصور الثابتة وعلى ذلك فإن تمثيل النظام بالإطارات المتلاحقة والقريبة من بعضها البعض زمنيا سوف يعطى التمثيل الدقيق للنظام، وبذلك يكون هناك وحدة للإطارات (Frame Axiom) والتي تعرف بأنها أصغر إطار يمكن إستخدامه، وأن تلاحق وحدات الإطارات يودى الى ابراز التغير الحادث.

(٧-٤-١) تصميم الإطارات

تتيح الإطارات تركيبة منتظمة من مجموعة خدمات لإظهار مجموعة من البيانات او قواعد البيانات بشكل سريع وواف، وتعتبر كذلك نظم عامة ذات خاصية إظهار القوائم (Lists) وتحتوى على فتحات (SLOTS) يمكن أن تستخدم لإظهار وتوصيف أكثر من قيمة، مثال ذلك إستخدام هذه الفتحات لأظهار قيمة معينة او إظهار حدود معينة او لتنشيط قاعدة معينة وهكذا يعتبر الغرض الاساسى لبناء الإطارات أنه يمكن أن تستخدم لتمثيل مجموعة كبيرة من الافكار والقواعد والحقائق، ويمكن للإطارات أن ترتبط فى نظام طبقي على أن يقوم كل اطار بتمثيل



شكل (٧-٤) التمثيل باستخدام الإطارات

مجموعة من الاهداف التى تتبع رتبة معينة للتركيب الطبقي، ومن مزايا الإطارات أنها يمكن أن تقوم بتمثيل معلومات طبقية، مثال ذلك المعلومات الإيضاحية والمعلومات الكيفية.

عناصر تصميم الإطارات: لتصميم الإطارات والتى تتبع التنظيم الطبقي وتورث الصفات طبقة الى أخرى فانه يلزم تعريف أربعة عناصر :

أ - العنصر الأول: الطبقة (CLASS) وهو التنظيم الطبقي الذى يسمح بتحديد مكان إطار معين وتحديد موضع ذلك الإطار من الإطارات الأخرى ونوعيته من حيث الرتبة فهل هو يتبع الطبقة العليا (UPPER CLASS) او هو فرع من طبقة عليا (SUB CLASS).

ب - العنصر الثانى: الوحدة (UNIT) والتى تنظم وجود الفراغات وعددها ومواصفات كل منها، وذلك لحفظ البيانات الايضاحية او الكيفية.

ج - العنصر الثالث: الفتحات (SLOTS) وهى التى تسمح باحتواء بيانات تشير الى قيمة او تورث لصفة ما.

د - العنصر الرابع: صفات التوارث (INHERITANCE) وهى التى تشير الى أن طبقة من الطبقات متورثة من طبقة أخرى من الطبقات العليا او السفلى، ولتوضيح ذلك فان شكل (٧-٥) يبين ذلك حيث يشير الى أن الطبقة يمكن أن تحتوى على طبقة أخرى ووحدة وفتحات وأن الوحدة يمكن أن تحتوى على كثير من الفراغات.

(٧-٥) التعرف على طرق تمثيل المعرفة

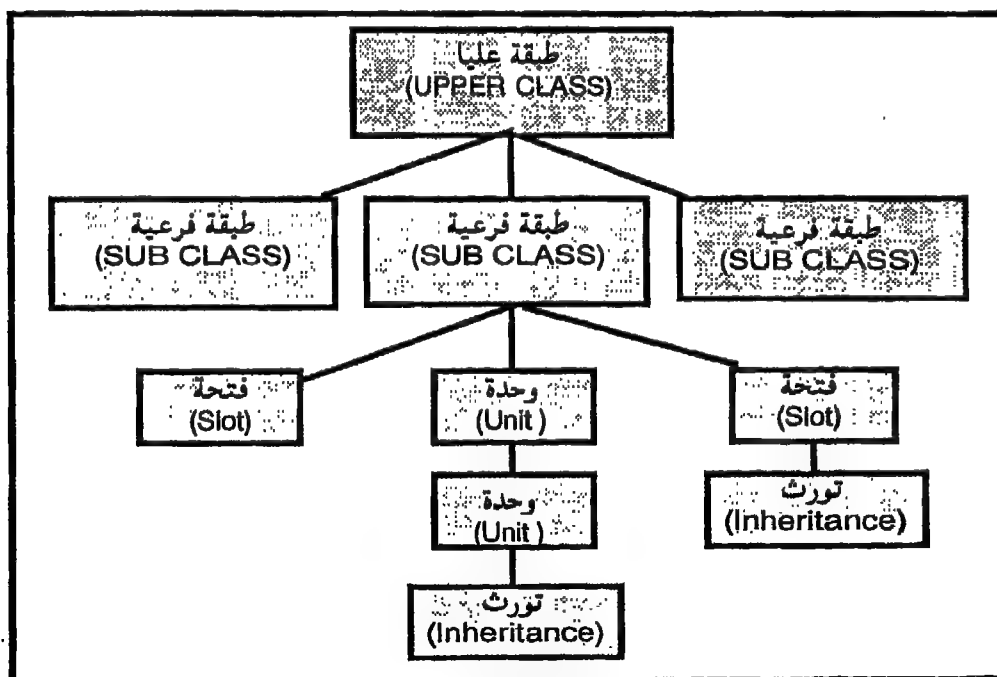
(Knowledge Representation Approaches)

تنقسم طرق التعرف على تمثيل المعرفة الى طريقتين :

١ - طرق التعرف الإعلانية (Declarative Approach) والتى يتم فيها تمثيل

كنه الحقائق (والتي تشمل التعرف على ماذا) (Knowing What) والتي تشمل على طرق التقارب المعتمدة على المنطق (Logic-based) والعلاقات والتي تؤدي الى الاستعانة بالتفرع الشجري (Trees)، والرسوم والشبكات الدلالية (Semantic Nets) وتعتمد كذلك على التوقع المنطقي الاسنادي (Predicate Logic).

٢ - طرق التعرف الاجرائية (Procedural Approach) والتي يتم فيها تمثيل طرق الفعل وكيفية التابع (التعرف على كيفية) (Knowing How) والتي تعتمد على قواعد النحو للغة (Grammars) والنظم الاجرائية المعتمدة على القواعد مثل نظم الإنتاج (Production Systems) والتي دائما ما تكون في الشكل " إذا توفر الشرط، يكون الفعل (Action) THEN (Condition) IF ويمكن توضيح الفرق بين التعرف الإعلاني والتعرف الاجرائي بالمثال الآتي: إذا نظرنا للجداول الرياضية التي تعطى لوغارتم الرقم، فيعتبر هذا النطاق نطاقا رقميا، ويصبح التعرف على لوغارتم الرقم تعرفاً إعلانياً أو وصفياً اما إذا اخذنا طريقة حساب اللوغارتم فإن ذلك يعتبر تعرفاً إجرائياً.



شكل (٧-٥) - تصميم الإطارات

الفصل الثامن

آليات تقنيات الاستدلال
(المنطق الرمزي الحسابي)

**Toolbox of Inference
Techniques
(Symbolic Computational
Logic)**

(٨-١) المنطق الرمزي الحسابي (Symbolic Computational Logic)

يعتبر المنطق الرمزي فرع من فروع الرياضيات والذي يمثل إحدى دعائم الذكاء الاصطناعي نظراً لقربه من مجال كثير من المشتغلين بعلوم وهندسة الحاسبات، حيث يعتبر صندوق الآلية لنظم الاستدلال. في هذا الباب يتم إلقاء الضوء على أسس المنطق الرمزي الحسابي التي تؤدي بدورها الى تفهم ميكانيكية المعالجة الرمزية والاستدلال.

من المعروف أن المنطق الرمزي يمكن أن يستخدم للتعبير عن جملة ما ثم الإستدلال والإستنتاج بجملة أخرى بناءً على الجملة الاولى. ويمكن كذلك الإثبات بوضوح عن صحتها، ويبدو للوهلة الاولى أن تطبيق المنطق الرمزي في أساليب الذكاء الاصطناعي سوف يكون مناسباً بدرجة كبيرة ولكن ذلك يشوبه التمييز الكامل وذلك لسببين :

١ - السبب الاول هو عدم إتجاه التصرف الإنسانى فى كثير من الأحيان للتعليل بناءً على الأسباب (Reasons) بأستخدام الإستدلال المنطقى (Logical Inference).

٢ - السبب الثانى هو عدم قدرة المنطق الرمزي على المرونة مما يجعله غير مناسباً فى التطبيق لبعض المجالات.

ولقد تطور إستخدام أساليب المنطق الرمزي الحسابي ليشمل :

١ - القاعدة الشرطية للتضمين الإيجابى (Modus Ponens) والتي

تساعد فى عمليات الإستدلال أو الإستنتاج (Inference) حيث يكون من المناسب

إستخدام القواعد المنطقية للاستدلال الحسابي.

٢ - النظم المعتمدة على القواعد (Rule - Based Systems).

٣- المنطق المتعدد القيم (Multivalued) والمنطق الغير متشابه التورية

(Non-monotonic) والمنطق المؤقت أو الإضطرابى (Temporal) وذلك

لتصميم نظاماً واقعية للذكاء الاصطناعي.

لتوضيح مفهوم كثير من التعابير المستخدمة فى المنطق الرمزي نورد التعاريف الآتية:

١ - الإستدلال أو الإستنتاج (Inference) تعرف عملية الإستدلال أو الإستنتاج بأنها

الإشتقاق لحقائق جديدة (Newfacts) من حقائق معروفة وتدخل فى ذلك النتائج

المطلوب إيجادها.

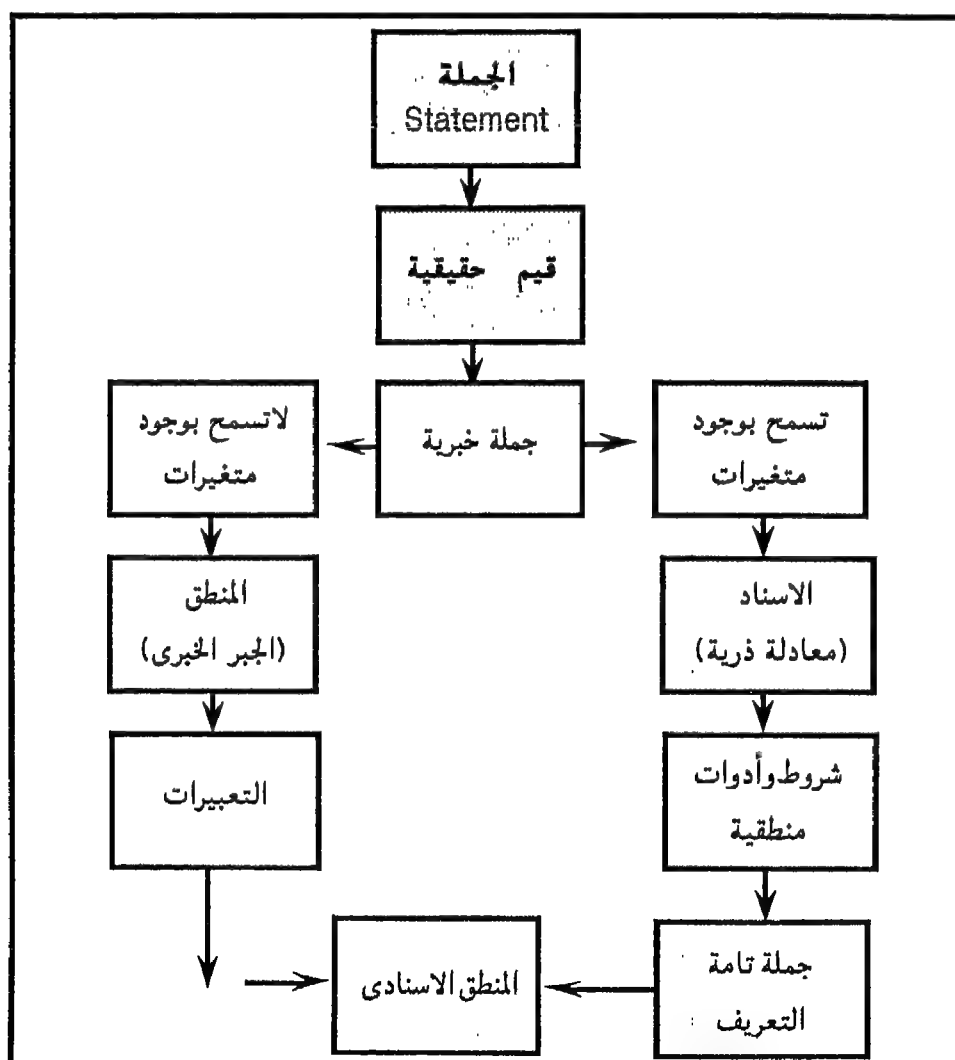
٢- الإشتقاق (Deduction) والذي يعرف بأنه تفاصيل عملية الإستدلال بإستخراج حقيقة

جديدة من حقيقة معروفة.

٣ - الإستنتاج الإستطرادى أو التأثيرى (Induction) ويعرف بأنه التعليل بالأسباب (Reasoning) لإستخراج حقيقة كاملة بمعرفة جزء من هذه الحقيقة أى أنه الوصول الى الكل من الجزء.

٤ - البديهية (Axiom) وهى الجملة الحقيقية المحتوية على الفرض الحقيقى أو الصحيح .
(١-١-٨) الجملة (Statement)

يستخدم المنطق الخبرى الرموز الرياضية لتمثيل الجمل وبذلك تعتبر الجملة الاعلانية أو البيانية إما حقيقة أو خطأ. ويبين شكل (٨ - ١) التمثيل الدلالى للجملة والذى يبين أن الجملة



شكل (٨-١) التمثيل الدلالى للجملة

الخبرية الصحيحة تنقسم الى قسمين :

١- القسم الأول :

يسمح بوجود متغيرات (Variables) لتكون جملة اسنادية (Predicate) تأخذ شكل المعادلة الذرية (Atomic formula) ، والتي تسمح بعد ذلك بإستخدام الأدوات والتراكيب المنطقية مثل الوصلات المنطقية لتعطي فى النهاية جملة فى شكل معادلة تامة التركيب والتعريف ((Well-formed formula(Wffs) والتي تكون فى النهاية المنطق الاسنادى أو الجبر الاسنادى (Predicate Calculus) .

٢- القسم الثانى :

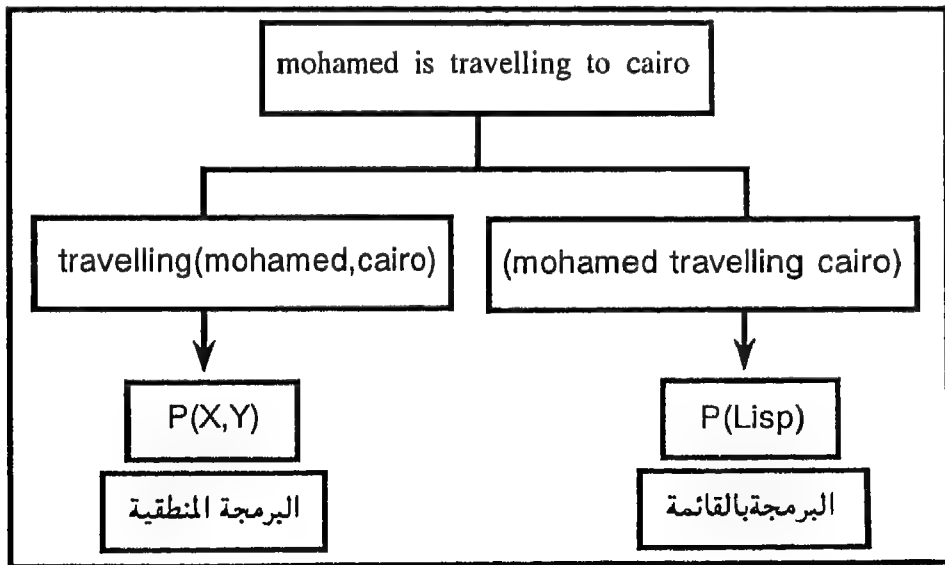
لايسمح بوجود متغيرات مع السماح بوجود الوصلات والتراكيب المنطقية لتعطي المنطق الخبرى أو الجبر الخبرى (Propositional Calculus) .

(٢-٨) التعبير الرمزى عن الجملة

يبين شكل (٢-٨) الجملة الانجليزية التى تشرح حقيقة سفر محمد الآن الى القاهرة والتى يمكن كتابتها و تمثيلها فى أحد الاشكال الآتية :

١- الشكل الأول :أسلوب القائمة (mohamed travelling cairo)

يأخذ شكل القائمة (List) المحتوية على ثلاثة عناصر ويكون من المناسب إستخدام لغة



شكل (٢-٨) التعبير الرمزى عن الجملة

البرمجة بأسلوب القوائم (Lisp) للمعالجة ويمكن تمثيل الجملة بالشكل الرمزي الذي يأخذ شكل الحرف (P) .

٢- الشكل الثاني: أسلوب المنطق (travilling(mohamed , cairo)
حيث تكون العلاقة الإسنادية (predicate relation) هي كلمة (travelling)
والتي تربط بين إسم الشخص محمد ومدينة القاهرة وبينهما أداة الوصل المنطقية (و)
ويكون من المناسب إستخدام لغة البرمجة المنطقية (Prolog) لهذا الشكل المنطقي.
ويمكن تمثيل الجملة بالشكل الرمزي الذي يسمح بوجود متغيرات فى شكل : $P(X,Y)$
حيث (X,Y) هي متغيرات.

الجملة المركبة (Compound Statement)
تتركب الجملة المركبة من عدة جمل بسيطة متصلة مع بعضها باستخدام أدوات الوصل
المنطقية مثل (و) (AND) (او) (OR) (لا) (NOT) وأداة الشرط (إذا) (IF) .

(٣-٨) قاعدة التضمن الشرطى (Implication)

تحتوى القاعدة الشرطية للتضمن على الأداة الشرطية (إذا) فى الشكل " إذا تحقق
الشرط تكون النتيجة " أو بمعنى ثان إذا تحقق الحدث يكون العمل كما يلى :
IF condition THEN consequent

حيث يمكن ان تكتب بالشكل الرمزي : $p \Rightarrow q$
والتي تعنى أنه إذا كان التمثيل الرمزي (P) صحيحا (True) فتكون النتيجة أو

p	q	$p \Rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

شكل (٣-٨) جدول الحقيقة

العمل (q) كذلك، ويمكن استخدام جدول الحقيقة شكل (٣-٨) (Truth Table) لتمثيل القاعدة الشرطية للتضمنين كما يلي :

		P	
		F	T
q	F	T	F
	T	T	T

شكل (٤-٨) خريطة كارنوف للتضمنين الشرطى $p \Rightarrow q$

والتي يمكن ان تبسط باستخدام خرائط كارنوف (Karnaugh Map) والمبينة فى شكل (٤-٨)، والتي تعطى الصورة العامة للتضمنين كما يلي :

$$p \Rightarrow q = (\text{Not } p) \text{ OR } q$$

والتي يجرى تفسيرها على أن التضمنين يكون صحيحا (T) فى أى الحالتين :
إذا كانت (Not P) غير صحيحة (F) أو (q) صحيحة (T).

(١-٣-٨) التضمنين الشرطى والنظم المبينة على القواعد

(Implication and Rule - Based Systems)

تعتبر قاعدة التضمنين الشرطى هى الأساس فى الإستدلال (Inference) فى النظم المبينة على القواعد، فإذا فرضنا أن التضمنين الشرطى ($p \Rightarrow q$) صحيحا (T) فإن هذه المعرفة يمكن إستخدامها فى الحكم على أن قيمة (q) تكون دائما صحيحة (T) طالما أن (P) ذات قيمة صحيحة (السطر الأول من جدول الحقيقة). وتحليل الجزء الثانى ومن قاعدة التضمنين ($(\text{Not } (P) \text{ OR } (q))$) والتي تحمل المعنى الإستدلالي : إذا كانت (P) قيمة حقيقية فإن (Not(P) تكون غير حقيقة، وعلى ذلك فإن قيمة (q) لابد أن تكون صحيحة. وتعتبر قاعدة التضمنين الشرطى هى الأساس فى التسلسل المتقدم (Forward Chaining) فى النظم المبينة على القواعد، حيث تكون النتيجة أو الفعل فى القاعدة الاولى مكون من عدة قواعد متسلسلة فى جمل مرتبطة بأداة المنطق (و) AND .

(٢-٣-٨) خواص قاعدة التضمنين الشرطى

تشتمل خواص قاعدة التضمنين الشرطى على ما يلى:

١ - إذا كانت كل من (p)، (q) قيماً لا تحتمل المعنى الصحيح فان التضمنين الشرطى

- ($P \Rightarrow q$) يظل دائما صحيحا (السطر الرابع من جدول الحقيقة) .
- ٢ - إن القاعدة الشرطية للتضمنين لاتعنى أن (p) تسبب (q) ولا يوجد علاقة سببية بهذا المعنى ولكن يوجد: أن (p) شرط كاف لوجود (q) ، أن (q) شرط مطلوب لوجود (p) .
- ٣ - يمكن اعتبار الجملة البسيطة قاعدة تضمنين مع عدم وجود الشرط (p) .
- ٤ - يمكن اشتقاق الآتى :
- أ - الوضع المقلوب ($q \Rightarrow P$) (Converse) .
- ب - الوضع المعكوس ($((\text{Not } P) \Rightarrow (\text{Not } q))$) (Inverse) .
- ج - الوضع المقلوب الإيجابي ($((\text{Not } q) \Rightarrow ((\text{Not } P)))$) .

(٤-٨) التسلسل المتقدم (Forward Chaining)

نفرض أن هناك مجموعة من القواعد التى تأخذ شكل جمل التضمنينات الشرطية-IF (THEN) يبلغ عددها (i) وأنه قد تم استخدام المتغيرات q_i, p_i فى التمثيل الرمزي للحقائق

لهذه القواعد بالترتيب الآتى

$$P1 \Rightarrow q1$$

$$P2 \Rightarrow q2$$

$$P3 \Rightarrow q3$$

فإذا فرضنا أن هناك تطابق بين النتيجة أو الفعل فى القاعدة الاولى (q1) مع الشرط فى القاعدة الثانية (P2) وكذلك كان هناك تطابق بين النتيجة أو الفعل فى القاعدة الثانية (q2) مع

الشرط فى القاعدة الثالثة (P3) كما يلى : $P2=q1$

$$P3=q2$$

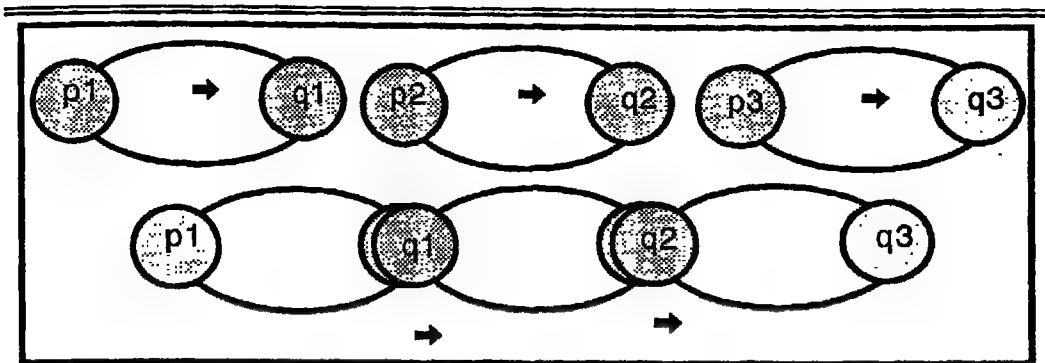
وينطبق ذلك على قاعدة المعرفة تصبح فى شكل سلسلة كما يلى :

$$P1 \Rightarrow q1$$

$$q1 \Rightarrow q2$$

$$q2 \Rightarrow q3$$

ويمكن تمثيل ذلك كما يلى فى شكل (٨-٥) حيث يجرى تمثيل كل قاعدة أو جملة من التضمنينات الشرطية بحلقة من سلسلة على طرفيها الحقائق (q , p) وعند تطبيق التساوى للطرف الثانى للحلقة الاولى (q1) مع الطرف الاول للحلقة الثانية (P2) . وتعميم ذلك بأن يتساوى الطرف الثانى من حلقة مع الطرف الأول للحلقة التى تليها حيث ينتج عن ذلك سلسلة تبدأ من الحلقة الأولى وتتفرع الى الأمام فى تسلسل متقدم كما هو مبين بالشكل .



شكل (٨-٥) التسلسل المتقدم

و لقد تم الإستعانة بالخطوات التالية :

١ - التحقق من أن جميع التضمينات الواردة بالقواعد الشرطية تشتمل على قيم حقيقية للمتغيرات (q_i) بناءً على قيم حقيقة للمتغيرات (P_i).

٢ - بناء تسلسل متقدم الى الأمام للوصول أو انتاج حقيقة جديدة (q_n) بناءً على معرفة الشرط الأول (P_1) ويمكن إستخدام القاعدة العامة التالية في بناء تسلسل متقدم.

- 1-(($p_1 \Rightarrow q_1$)AND p_1 being True) \Rightarrow (q_1 is True)
- 2-((q_1 isTrue)AND($q_1=p_2$)being True Yields p_2 is True)
- 3-(($p_2 \Rightarrow q_2$)AND p_2 being True) \Rightarrow (q_2 is True)
- 4-(q_2 isTrue)AND($q_2 = p_3$)being True Yields p_3 is True
- 5-(($p_3 \Rightarrow q_3$)AND p_3 being True) \Rightarrow (q_3 is True)

(٨-٥) شبكات الإستدلال (Inference Nets)

يمكن بناء شبكة الإستدلال للمثال السابق وذلك كما هو مبين بالشكل (٨-٦) حيث يرسم سهم من رأس القاعدة (الشرط) الى ذيل القاعدة (النتيجة أو الفعل) والذي يعرف بأنه إطلاق للقاعدة (Rule firing) وإستخدام تقرب للمراعاة الرمزية (Symbolic Matching Approach) لعقد المساواة بين حقائق (النتيجة أو الفعل) لقاعدة مع الشرط للقاعدة التي تليها، حيث \times قتل موازنة رمزية ، وتصبح النتيجة :

$p_1 \Rightarrow q_3$
 given deduced

(٦-٨) قاعدة التضمن الشرطى الإيجابى (Modus Ponens)

تستخدم قاعدة التضمن الشرطى الإيجابى (MP) للإشتقاق (Deduction) فإذا فرضنا أن هناك بديهة حقيقية (P) وكان التضمن الشرطى $(P \rightarrow q)$ صحيحاً دائماً فإن النتيجة أو الفعل لابد أن يكون صحيحاً ويمكن كتابة هذه القاعدة كما لآتى :

P is True

$P \rightarrow q$ is True

q conclusion is True

$(P \text{ AND } (P \rightarrow q)) \rightarrow q$ أو

(٧-٨) الإسنادات والمتغيرات (Predicates and Variables)

إزالة الخلط بين إستخدام الإسنادات (predicates) والدوال الإسنادية (Predicate Functions) يمكن توضيح التعريفات الآتية :

١ - الإسناد (Predicate) هو جملة خبرية تحتوى على متغيرات (Variables) .

٢ - الدالة الإسنادية (Predicate Function) هى الدالة التى تحدد القيم الصحيحة أو الخطأ

لإسناد ما. مثال ذلك : يمكن كتابة الإسناد للأرقام الزوجية كمايلى : $\text{Peven}(n)$

وبذلك تكون الدالة الإسنادية التى تحدد قيم المتغيرات لهذا الاسناد هى الدالة :

$\text{Peven}(n) = \begin{matrix} \text{True if } n \text{ is even} \\ \text{False otherwise} \end{matrix}$

حيث تصبح قيمة الرقم الزوجى حقيقية.

(٨-٨) المنطق الإسنادى (Predicate Logic)

يعرف المنطق الإسنادى بأنه فرع من فروع المنطق الذى يسمح بنمذجة (Modelling) الحقيقة فى جملة ما إعتماًداً على القيم التى تم فرضها فى جزء من الجملة

قاعدة ١	x	قاعدة ٢	x	قاعدة ٣
$p1 \rightarrow q1$	\rightarrow	$p2 \rightarrow q2$	\rightarrow	$p3 \rightarrow q3$
معطاة	إشتقاق	إشتقاق	إشتقاق	إشتقاق

(شكل ٦-٨) شبكة الإستدلال

كما تعرف مجموعة القيم التي تحدد المتغيرات في إسناد ما على أنه المجال الكلى (أو عالم) لهذا المتغير (Universe) وبذلك يمكن القول بأنه :

" يتساوى إسنادان إذا كان لكل منهما نفس القيم الحقيقية التي تحدد المتغيرات فيها "

(٨-٨-١) الإسنادات المعتمدة على الأرقام (Numerically Based Predicates)

$$y^2 - 3 = 6$$

إذا فرضنا المعادلة الجبرية الآتية :

والتي يمكن أن تمثل أحد الإسنادات المحتوية على المتغير (y) وبذلك يكون المجال الكلى

لهذا الاسناد مساوياً " لمجموعة من الأرقام الحقيقية، وعند اختيار القيم $y = -3$, $y = 3$ فإن هذا الإسناد يكون صحيحاً، ومن الممكن كتابة الإسناد بطريقة أخرى مكافئة كمايلي :

$$(y + 3) (y - 3) = 0$$

وإذا كان المجال الكلى (Universe) للمتغيرات في دالة إسنادية محدداً فإن هذه الدالة

يمكن أن تأخذ شكل جدولى (Tabular Form) .

(٨-٨-٢) الإسناد ذو المتغير الواحد (Unary Predicate)

تستخدم الإسنادات ذات المتغير الواحد لتعريف علاقة واحدة مثال ذلك، إذا أردنا أن نوصف

العلاقة الاسنادية إمكانية التحرك (may move) التي ترتبط بالاشياء المتحركة كمايلي :

may_move (Object)

فإن (may_move) هو الإسناد والشئ المتحرك (Object) هو المتغير الواحد التي

يمكن ان يأخذ قيما مختلفة كمايلي :

Object = { man , horse , car , mountains }

من الواضح أن العلاقة الإسنادية (may_move) تكون ذات قيما حقيقية (True)

في الثلاث حالات الأولى (man, horse, car) وخاطئة (False) في الحالة الأخيرة (mountains).

ويمكن إستخدام الإسناد ذو المتغير الواحد في بناء إسنادات ذات متغيرات متعددة

(n_place predicate) كما هو مبين في العلاقة الثنائية التي تصف أن الشئ الأول

(Object 1) يقع لاحقاً من (next_to) الشئ الثانى (Object 2) كمايلي :

next_to(Object 1, Object 2) .

علماً بأن المتغيرات (Object 1, Object 2) يمكن أن تأخذ قيما مختلفة.

أو يمكن أن تأخذ الشكل :

on_top_of(Support, Object)

لوصف مجموعة من الدعائم (Support) التى تحمل فوقها مجموعة من الاشياء أو الكيانات (Objects) بحيث تحمل كل دعامة أحد الأشياء .

(٣-٨-٨) اختيار التمثيل للإسنادات (Choosing Predicate Representation)

إذا فرضنا أن هناك دائرة كهربية تحتوى على أربعة خطوط (lines) لنقل التيار الكهربى ، وان كل خط من الخطوط مزود بقاطع للدائرة (circuit - breaker) والذي يتولى فتح الدائرة لقطع التيار عند الحاجة لذلك أو عند زيادة التيار زيادة كبيرة فوق القيمة المسموح بها كما فى شكل (٧-٨) وأردنا ان نقوم بتشفير (Coding) وتمثيل الجملة التى توضح أن القاطع المركب على الخط الرابع مفتوح :

(the circuit-breaker in line 4 is open)

بسلسلة من الإسنادات المختلفة الشكل كالآتى :

١ - الشكل الأول : is_open (circuit-breaker,line 4)

٢ - الشكل الثانى : circuit-breaker (line 4 , open)

٣ - الشكل الثالث : line 4 (circuit-breaker , open)

والتي يمكن تفسيرها من خلال المنطق الإسنادى بالعلاقات الثنائية الآتية :

is_open(What,Where) ,
circuit-breaker(Where,Status) ,
line 4(Protection , Status) .

حيث تشتمل المتغيرات على مايلى :

What = { circuit-breaker,disconnect - switch,line-itstef}

Where = { line 1 , line 2 , line 3 , line 4 }

Status = { open , close }

Protection = { circuit - breaker , relay , recloser }

وغالبا يجرى تعريف وتشفير العلاقات بإستخدام الفعل أى فى الشكل الأول.

(٤-٨-٨) الإسنادات والجمل المركبة (Predicates and Compound Statement)

يمكن إستخدام الإسنادات لتكوين جمل مركبة كمايلى :

إذا فرضنا أن علاقة الأبوة تمثل رمزيا فى الشكل (P) وان الأب (X) والأبن (Y)

وكذلك الأب (Y) والأبن (Z) فانه يمكن كتابة العلاقات الاسنادية الآتية :

P(X,Y)

P(Y,Z)

فإذا فرضنا أن العلاقة الإسنادية بين الجد والحفيد هي (S) ، فإنه يمكن كتابة الجملة المركبة الآتية والتي تأخذ شكل قاعدة (Rule) للتضمنين الشرطى والتي تفيد أن : " إذا كان (X) هو الأب للشخص (Y) وأن الشخص (Y) هو الأب للشخص (Z) ، فتكون النتيجة أن (X) هو الجد للشخص (Z) " ، وتكتب كما يلى :

$$P(X,Y) \text{ AND } P(Y,Z) \Rightarrow S(X,Z).$$

(٨ - ٩) محددات الكمية (Quantifiers)

للتقدير والتحديد الكمي يمكن إستخدام محددات الكم (All , Some) كما هو مبين فى المثال التالى :

إذا فرضنا الجمل الآتية :

All (A) are (B) ,
Some (A) are (C) .

حيث توضح مفهوم الجملة الأولى أن كل (A) يكون (B) وتوضح الجملة الثانية أن بعض (A) يكون (C) وبذلك يمكن إستخراج العلاقات الآتية :

- 1- All (every) (A) is a (B)
- 2- No (A) is a (B)
- 3- Some (at least one) A is not (C)
- 4- Some (at least one) A is (C)

وبذلك يمكن تعريف (All), (No) كمحددات كمية كلية (Universal Quantifiers) وكذلك (Some) كمحدد جزئى (Existential Quantifiers) والتي تعنى أنه على الأقل يوجد واحد فى هذه الكمية له الصفة (C) ، ويمكن شرح عمل محددات الكمية فى المثال التالى : إذا فرضنا أننا نحاول أن نكون قاعدة بيانات رمزية لوصف الشئ المتحرك بأنه لا يمكن أن يرى ثابتا والتي تأخذ الشكل الآتى :

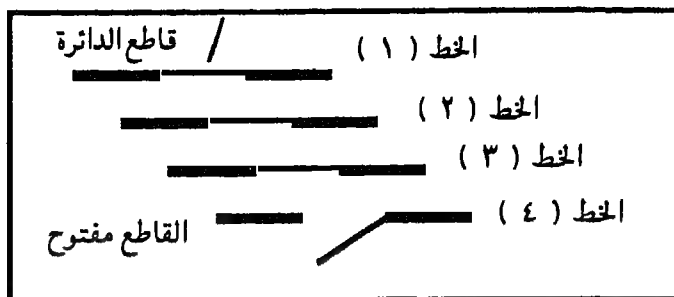
" Any object that is observed to be moving
cannot be stationary "

والتي يمكن كتابتها فى شكل قاعدة للتضمنين الشرطى كما يلى :

observed-moving(Object) \Rightarrow Not Stationary (Object)

والتي تعنى الحقيقة أن جميع الكيانات المتحركة لا يمكن أن تكون ثابتة. ويمكن كتابة محددات الكمية الكلية فى الشكل الآتى :

(AX) p(X)



شكل (٧-٨) دائرة كهربية تحتوى على قواطع للتيار

حيث يمكن نطقها كما يلي :

" لكل قيم (X) فإن الإسناد (P) يكون صحيحاً "، ومن الواضح أن العلاقة الإسنادية (P) لا تعتمد على قيمة المتغير (X) والتي تعنى كذلك أن الجملة تكون صحيحة لكل قيم المتغير. وتختلف الصورة بالنسبة لمحددات الكمية الجزئية والتي تعنى أنه على الأقل توجد قيمة واحدة تجعل الاسناد صحيحاً والذي يمكن كتابته في الصورة الآتية: $p(X) : (SX)$ حيث يمكن قراءته كما يلي : " لبعض قيم (X) يكون الإسناد صحيحاً " أو توجد قيمة واحدة للمتغير (X) تجعل الاسناد (P) صحيحاً. وتستخدم محددات الكمية في ربط المتغير بالدوال الإسنادية كزوائد أولية.

(٨-٩-١) تكافؤ الإسنادات المحددة الكمية

(Equivalence of Quantified Predicates)

يمكن عقد التكافؤ بين الإسنادات المحددة الكمية باستخدام أداة النفي (NOT) كما هو مبين في المثال التالي :

$$\text{NOT} ((A \ X) P(X)) = (SX) \text{NOT} P(X).$$

$$\text{NOT} ((SX) P(X)) = (AX) \text{NOT} P(\bar{X}).$$

ويمكن الحصول على جمل تحتوى على محددات الكمية مع قاعدة التضمين

$$(AX) P(X) \Rightarrow P(a)$$

الشرطى في الشكل التالي :

حيث (a) هو المجال الكلى للمتغير (X) . ويمكن كتابتها بطريقة اخرى كما يلي :

$$P(a) \Rightarrow (SX) P(X)$$

(٨-٩-٢) التكرار باستخدام النفي (Tautology)

عند إثبات النظريات يمكن استخدام الإعادة المتكررة للإسناد نفسه باستخدام النفي فمثلا

$$(NOT) P(OR)$$

يمكن كتابة :

والتي تمثل تكرراً يكون حقيقياً دائماً بصرف النظر عن قيمة الإسناد (P)، كما يمكن

P(AND) (NOT) P

كتابة :

- تكون خطأ دائماً حيث تمثل التضاد أو العكس لشيء واحد. وما سبق توضيحه من فروض تعتبر الأساسيات للإطار العام للرياضيات المنطقية والتي يمكن إستخدامها فيما يلي :
- ١- الحصول على أو إنتاج حقائق جديدة من حقائق معرفة .
 - ٢- اثبات صحة أو عدم صحة فرض معين .
 - ٣- اثبات نظرية ما .

وبذلك يمكن القول بأن القدرات الحسابية المنطقية الرمزية هي صندوق الآليات لتقنيات الإستدلال (Toolbox of Inference Techniques)

(٣-٩-٨) اللغة المنطقية أحادية الرتبة

- لإستخدام اللغة المنطقية أحادية الرتبة يمكن كتابة التعريفات الآتية :
- ١ - يعرف الحد (Term) بأنه القيمة الثابتة أو المتغيرة أو الدالة المحتوية على عدد من المتغيرات.
 - ٢ - تعرف الدالة (Function) بأنها مجموعة من الحدود التي تبدأ بدليل (Functor) والذي غالبا ما يكون رمزا أو اسماً و الذي يدل على معنى هذه الدالة .
 - ٣ - يعرف الإسناد على أنه مجموعة من الحدود المسبوقه برمز إسنادي كسمى .
 - ٤ - باستخدام أدوات الوصل المنطقية وتطبيق الشروط على هذه الحدود والدوال والاسنادات فإنه يمكن الحصول على المعادلات التامة التركيب (Well - formed formula) وتعرف المعادلات التامة التركيب (Wffs) بأنها جمل أو إسنادات يمكن أن تأخذ الأشكال الآتية :

NOT a
a AND b
a OR b
a → b
b → a
Q (a)

- حيث (a) , (b) يمكن أن تكون جملاً أو إسنادات أو معادلة أولية وتمثل (Q) محددات الكمية، وبذلك فإن المعادلات التامة التركيب تعتمد على الآتى :
- أ- تعدد الإسنادات (N-Place Predicate) .
 - ب- الجمل البسيطة (Simple Statement) .
 - ج- النفي (Negation) .

د - الضم (Conjunction) .

هـ - الفصل (Disjunction) .

و - التضمين الشرطى (Implication) .

ز - التحديد الكمى (Quantification) .

تعرف الجملة أو المعادلة المقفلة (Closed Formula) بأنها الجملة التى لا تحتوى على متغيرات حرة، فإذا احتوت الجملة على متغيرات حرة فإنها تصبح تعبيراً (Expression) وذلك يمكن تعريف لغة المنطق إحادى الرتبة كما يلى :

" إذا تم إستخدام المجال الكلى للمتغيرات لإستبعاد الدوال والاسنادات لإشتقاق جمل جديدة، فإن هذه الجمل الجديدة تعتبر ممثلة بلغة المنطق الأحادى الرتبة " .

(٨-١٠) إيجاد الحلول للمشكلات (Problem Solving)

كما رأينا سابقا كيف يمكن إستخدام قاعدة التضمين الشرطى الإيجابى (MP) لتصميم قواعد البيانات المعتمدة على القواعد، وفيما يلى يتم دراسة إستخدام المنطق فى بناء ميكانيكية تستخدم لإيجاد الحلول للمشكلات بشكل عام. ومن أهم الموضوعات ما يلى :

١- الإثبات بالتحليل (Resolution) .

٢- التوحيد (Unification) .

(٨-١٠-١) الإثبات التحليلى أو التفنيدي (Proof by Resolution (Refutation))

يمكن تعريف الإثبات بإستخدام طرق التحليل أو طرق التفنيدي بأنه إستخراج بنود أو تعبيرات (clauses) جديدة من بنود أولية، وفى حقيقة الأمر فإن المنطق يساعد على الحصول على بنود جديدة قائمة بذاتها. ويعتبر الإثبات بإستخدام طرق التحليل التفنيدي الأداة القوية والطرق الغير مباشرة لبناء ميكانيكية الإستدلال.

إن إستخدام أدوات الإثبات بطرق التحليل فى عملية الإستدلال تعتبر من عمليات محاكاة الدهاء (Subtle simulation) ، وكثيرا ما تستخدم هذه الطرق لإثبات أن مجموعة من البنود غير معتمدة على نفسها، وذلك بإستنتاج أو إشتقاق التضاد أو العكس المنطقى، بشرط أن تكون البنود فى شكل معادلات، كما يمكن إثبات أن بند من البنود يمثل حقيقة ما إذا كان هذا البند ضمن مجموعة من البنود التى تعرف بأنها حقيقية، وبإستخدام أداة المنطق للنفي لا (NOT) والبحث عن العكس أو التضاد فإذا كانت النتيجة السالبة التى تم إشتقاقها غير

متوافقة مع قاعدة البيانات، فإن هذا البند لا يمكن أن يكون صحيحا . مثال ذلك إذا فرضنا أن القيمة (a) تمثل حقيقة وأن نفي هذه الحقيقة تعتبر حقيقة كذلك فإن التضاد (Contradiction) يصبح فى الشكل :

(a)AND NOT(a)

لا بد أن يكون صحيحا ، ولتنفيذ ذلك تتبع الخطوات التالية :

- أ - ترتيب بنود قاعدة البيانات فى شكل معادلات تامة التركيب (Wffs) .
- ب - فرض النفي لحقائق قاعدة البيانات ووضعها فى الترتيب مع قاعدة البيانات الأولى.
- ج - البحث عن التضاد (Contradiction) لإثبات صحة الفرض .
- ولوضع نظام إجرائى (Systematic Procedure) يتبع ما يلى :
- ١ - تحويل الجمل (Statement) الى قاعدة بيانات منطقية (logical d. base)
- وتحويل الفروض (Hypothesis) الى تعابير أو بنود (Clauses) .
- ٢ - تحويل الجمل التامة التركيب والمحتوية على قواعد التضمنين الشرطى ومحددات الكمية الى شكل يسهل تحليله كما يلى :

أ- التضمنين الشرطى $(p \Rightarrow q)$ تحول الى الشكل :

$((NOT P) OR q)$

ب- التكافؤ أو التساوى $(p = q)$ تحول الى الشكل :

$((NOT P) OR q) AND (p OR (NOT q))$

٣- تحليل البنود وذلك بإختيار كل بندين لإختصارهما الى بند واحد كما يلى :

a OR b	T
a OR c	T
b U c	T

والتي تعنى أنه إذا كانت (a) أو (b) حقيقة وكانت كذلك (a) أو (c) حقيقة فإن (b) أو (c) تكون حقيقة كذلك. والتي يمكن أن تأخذ الشكل الأتى :

$(a OR b) AND (NOT a OR b) \Rightarrow (b OR c)$

والتي تأخذ شكل قاعدة التضمنين الشرطى والتي تعنى أنه إذا توفر الشرط (a) أو (b) ونفى (a) أو (b) فتكون النتيجة (b) أو (c) .

مثال للاستدلال باستخدام التحليل: نفرض أن هناك قاعدة بيانات أولية (D1) والتي تحتوى على جمل صحيحة ومعتمدة على نفسها والتي تأخذ الشكل التالى :

1- P 1

2- P 1 \Rightarrow q1

3- q 1 \Rightarrow q 2

ويصبح إحراز او اصابة الهدف (Goal) هو إثبات أن (q2) صحيحة باستخدام قاعدة البيانات (D1). ولإثبات ذلك تتبع الخطوات التالية :

أ - اضافة نفى (q2) فى شكل NOT q2 الى قاعدة البيانات (D1) لتصبح بعد التعديل (D2) كمايلي :

1- P 1

2- NOT P1 OR q1

3- NOT q1 OR q2

4- NOT q2

ومن (٢) ، (٣) يمكن إستخلاص أن :

q1 OR NOT p1

NOT q1 OR q2

NOT p1 OR q2

ب- وبإضافة البند الناتج من العملية السابقة الى قاعدة البيانات D2 تنتج قاعدة البيانات D3 كالآتى :

1- p1

2- NOT p1 OR q1

3- NOT q1 OR q2

4- NOT q2

5- NOT p1 OR q2

ومن (٤) ، (٥) يمكن إستخلاص الآتى :

NOT q2

q2 OR NOT p1

NOT p1

ج- وبإضافة البند الناتج من العملية السابقة الى قاعدة البيانات D3 تنتج قاعدة البيانات D4 الآتية :

1- p1

2- NOT p1 OR q1

3- NOT q1 OR q2

4- NOT q2

5- NOT p1 OR q2

6- NOT p1

د - من (1) ، (6) يمكن إستخدام معادلة التضاد (Contradiction) كمايلي :

NOT p1 AND p1

وبالحصول على معادلة التضاد فإن هذا يعنى أن إضافة النفي (NOT q2) الى قاعدة البيانات (D1) لا يعطى نفس النتيجة (Inconsistent) وبذلك تكون (q2) حقيقة (True) ويمكن القول بأنه إذا تم إضافة النفي ولم نحصل على معادلة التضاد فإن ذلك يعنى أنه خطأ والتي تشير الى عدم صحة الفرض .

(٨-١١) التوحيد (Unification)

التوحيد هو منظومة إجرائية تستخدم لتحديد قيم المتغيرات فى الجمل والإسنادات والمعادلات التامة التركيب (Wffs) . حيث أن القيم الحقيقية للإسنادات تعتبر دالة فى القيم الموصفة لهذه الإسنادات ولذلك فإن إيجاد القيم للمتغيرات سوف تثبت حقيقة هذه الجمل أو الاسنادات. ويعتبر التوحيد عملية أساسية فى تصميم ميكانيكية الإستدلال فى الذكاء الاصطناعى وتعتبر كذلك القلب المحرك للغة البرمجة المنطقية، وتعتمد عملية التوحيد على تنظيم عملية التعويض (Substitution). ويمكن تعريف عملية التوحيد بأنها محاولة إثبات أن التعبيرين متساويين وذلك بإيجاد القيم المناسبة للمتغيرات بالتعويض أو الربط التى تجعل التعبيرين متساويين، كما يعرف التعويض بأنه التحديد لقيم المتغيرات بحيث لا يحصل متغير على قيمتين مختلفتين، وبذلك يمكن إتباع القاعدة الآتية :

" تعتبر مجموعة من التعبيرات أو البنود متوحدة (Unifiable) إذا توفر الشرط القاطع (if and only if) بأن يكون هناك تعويض واحد على الأقل يجعل هذه التعبيرات متشابهة أو متساوية ". وعند إستخدام التعويض السابق لمرحلة التوحيد فإنه يمكن إحلال المتغير بقيمة ثابتة أو بقيمة متغيرة أو بدالة لاحتوى على متغيرات وذلك لتجنب حدوث الدخول فى تسلسل دائرى.

(٨-١١-١) قواعد التوحيد (Unification Rules)

يمكن إستخدام القواعد الآتية فى عملية التوحيد :

- أ - يتوحد عنصران (ذرتان) فى شكل ثوابت أو متغيرات إذا تشابها .
- ب - تتوحد قائمتان أو تعبيران (يمكن تحويلهما الى قائمتين) إذا تشابها . ج- يتوحد الثابت مع المتغير الغير محدد على قيمة بحيث يتم ربط المتغير بالقيمة المحددة للثابت .
- د- يتوحد متغير غير محدد مع متغير محدد القيمة وذلك بإحلال التحديد للمتغير الأول.

- هـ- يتوحد متغير محدد القيمة مع قيمة ثابتة إذا كانت القيمة المحددة مشابهة للقيمة الثابتة.
- و- يتوحد متغيران غير محددي القيمة إذا تم لأحدهما تحديد القيمة في مرحلة سابقة حيث يأخذ المتغير نفس القيم المحددة للمتغير الأول.

(8-11-2) التعويض والربط (Substitution and Binding)

قبل إجراء عملية التوحيد يلزم أولاً التعويض وربط قيم للمتغيرات مثال ذلك : إذا تم إحلال الثابت (C1) محل المتغير (V1) والتي تكتب في الشكل (C1/V1) والتي تعنى أنه تم ربط المتغير (V1) بقيمة الثابت (C1) ، وإذا تم كذلك إحلال المتغير (V1) محل المتغير (V2) والتي يمكن أن تكتب في الشكل (V1/V2) فإن مجموعة التعويضات (S) (Substitution Set) سوف تكون في الشكل :

$$S = (C1/V1) , (V1/V2) = (C1/V2)$$

فإذا فرضنا أن هناك جملة ما وتم إستخراج مجموعة التعويضات (S) في الشكل :

$$S = (ti/Ri) , i = 1, 2, 3, \dots, n$$

حيث يتم إحلال (Ri) محل (ti), فإذا فرضنا كذلك أن هناك جملتين (P2, P1) ويراد توحيدهما، ولتنفيذ ذلك يتم إستخراج مجموعة التعويضات لكل منهما (S1), (S2) .

وتصبح عملية التوحيد تامة إذا كانت مجموعة التعويضات الأولى (S1) مساوية لمجموعة

$$P1S1 = P2S2 \quad \text{التعويض الثابتة (S2) أى أن :}$$

$$\text{IF } S1 = S2$$

$$\text{THEN } P1 = P2$$

وبذلك يمكن إقرار القاعدة التالية : تتوحد جملتين إذا كانت مجموعة التعويض للمتغيرات (S) لكل منهما مساوية للأخرى.

(8-11-3) خوارزميات التوحيد (Unification Algorithms)

لفهم عملية التوحيد وإشتقاق الخوارزميات لها فان ذلك يتضح في المثال التالي :

نفرض أن هناك جملتان (a), (b), تامتان التركيب (Wffs) وأن هذه الجمل يمكن كتابتها

في شكل قوائم لتصبح القائمة الأولى (List-a) والقائمة الثانية (List-b) . ويراد

تصميم الخوارزميات (U1), (U2) لإجراء عملية التوحيد بين القائمتين على أن :

١ - تكون النتيجة النهائية هي الحصول على قائمة التعويض أو قائمة الربط (S) كقائمة صفرية

(Nil) أو قائمة خالية (Empty) وذلك في حالة تساوى وقائيل القائمتين، ولجاح التوحيد.

٢ - إذا لم تتم عملية التوحيد فإن النتيجة تؤدي الى عدم النجاح (FAIL) .

٣ - تكون طبيعة الخوارزم المستخدم تكرارية (Recursive) وأن تتم عملية التوحيد على أساس عنصرا بعنصر (Element by element) ويبين الشكل (٨-٨) خرائط التدفق (Flow Chart) للخوارزم (U1) والذي يتكون من جزئين D1,D2 حيث يكون المطلوب التوحيد بين القائمتين.

ويمكن كتابة الخوارزم (U1) باستخدام تعبيرات لغة الباسكال كمايلي :

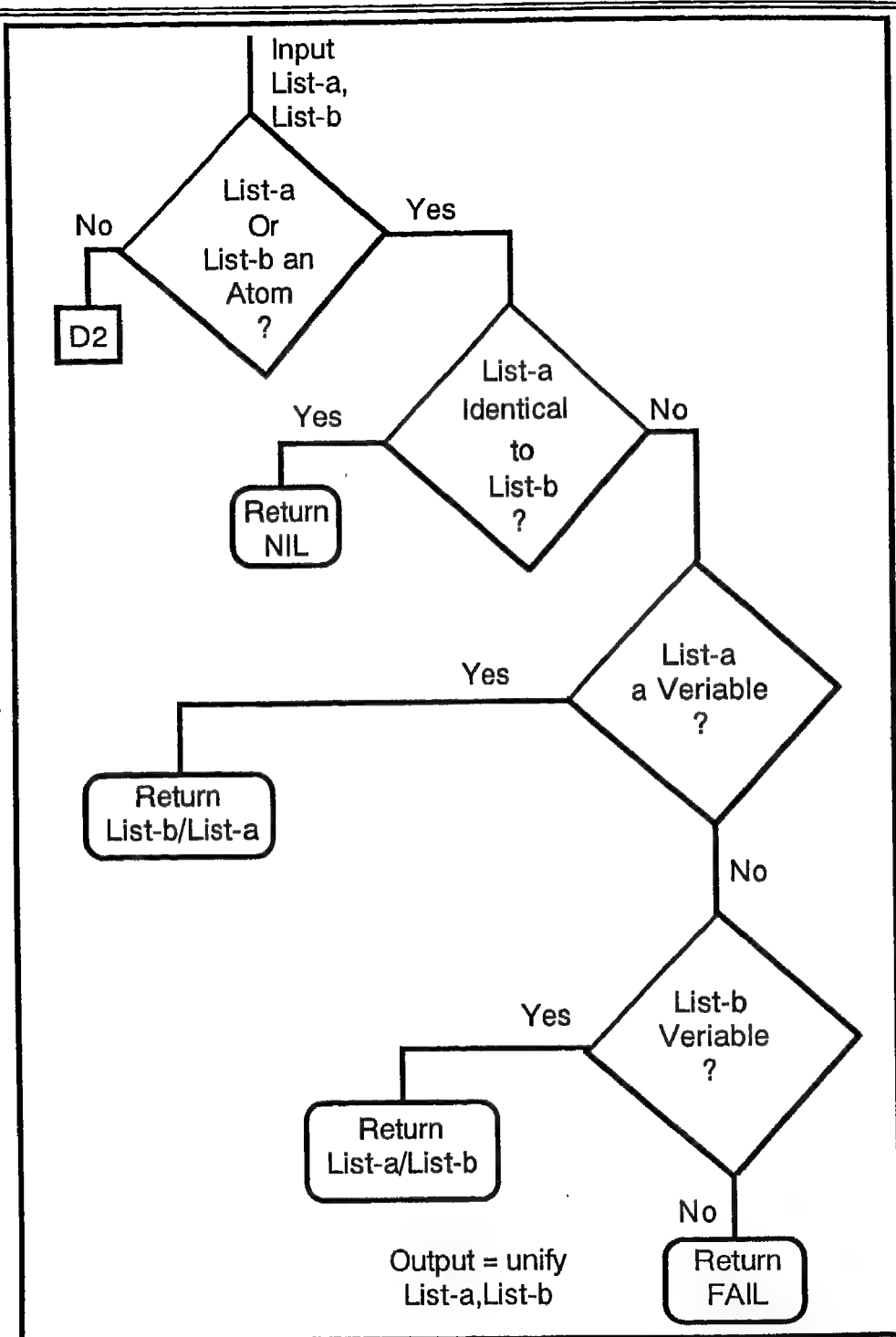
```

PROCEDURE UNIFY1 (List1,List2);
BEGIN
  IF List-a or List-b consist of a single element (an atom)
  THEN
    IF List-a and List-b are identical
    THEN return empty list, NIL {no substitutions}
    ELSE
      IF List-a is a variable
      THEN return (List-a/List-b)
      ELSE IF List-b is a variable
      THEN return (List-a/List-b)
      ELSE return FAIL;
  ELSE { not atoms ; must be in lists }
  BEGIN
    Head1 :=1st element of List-a ;
    Rest1:=rest of List-a;
    Head2:=1st element of List-b;
    Rest2:=rest of List-b;
    M1:=UNIFY1 ( Head1 , Head2);
    IF M1 = FAIL
    THEN return FAIL
    ELSE
      BEGIN
        Rest1:= apply (M1,Rest1) ; apply substitution M1
        Rest2:= apply (M1,Rest2);to Rest1 and Rest2
        M2 := UNIFY1 (Rest1 , Rest2);
        IF M2 = FAIL
        THEN return FAIL
        ELSE return composite of M1 and M2
      END
    END;END; { UNIFY }
  
```

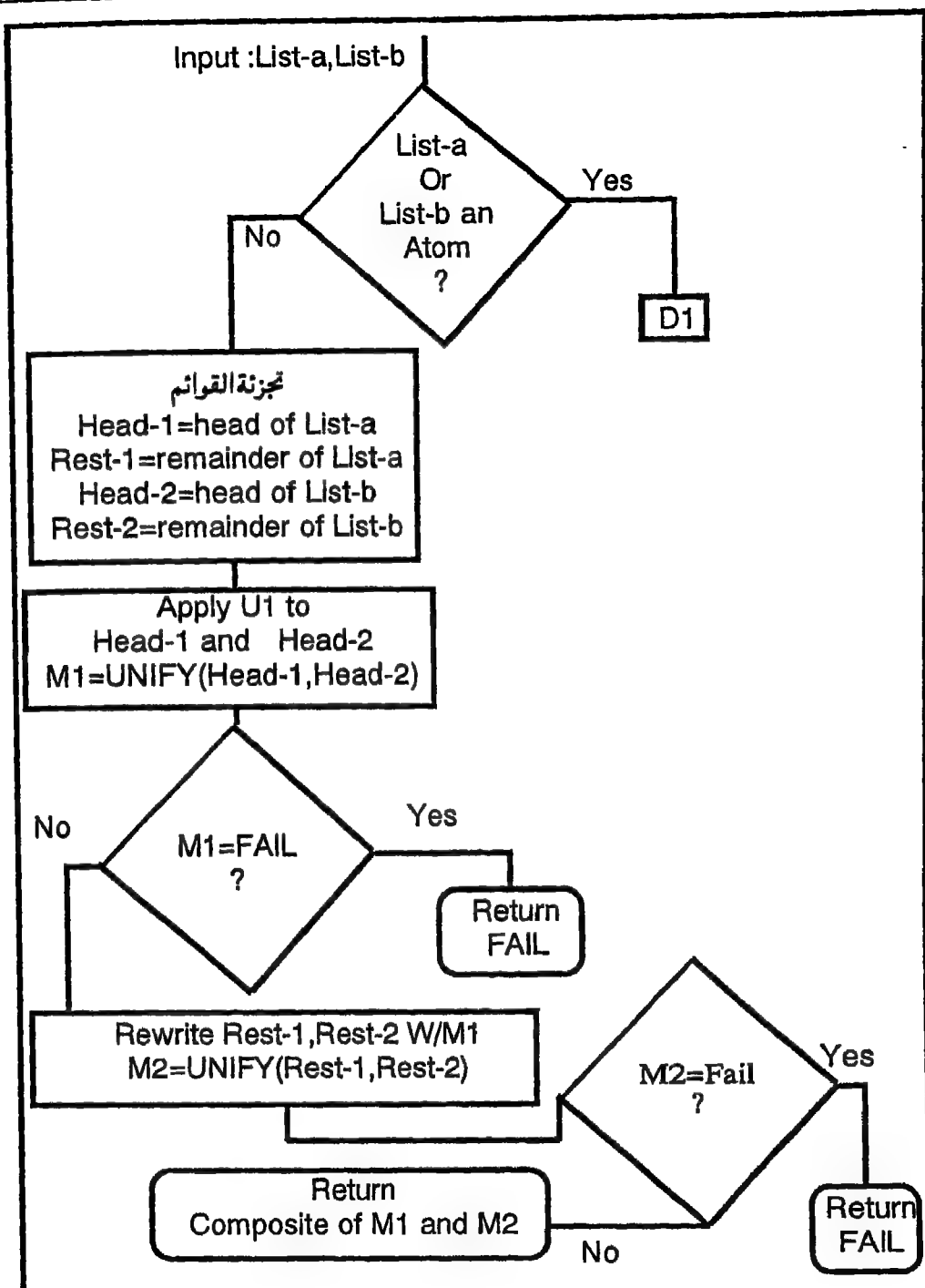
من خريطة التدفق شكل (٨-٨) وشكل (٨-٩) يمكن وضع الملاحظات التالية :

١ - يتم إتخاذ التوحيد بين عنصرين أوليين (Single atomic element) كأساس لبناء الموحد مع إمكانية تطبيقه بعد ذلك على القوائم المركبة من أكثر من عنصر وذلك بتجزئتها بالتعاقب (Successive Substitution) الى عناصر أولية يسهل إجراء عملية التوحيد عليها وإستخدام التكرار فى ذلك.

٢ - يتفرع الموحد (U1) الى فرعين طبقا لعدد العناصر فى القوائم فإذا كانت القائمتان ذاتا



(شكل ٨-٨) خريطة التدفق للجزء الأول (D1) من الموحد (U1)



شكل (٨-٩) خريطة التدفق للجزء الثاني (D2) من الموحد (U1)

- عنصر واحد فاننا نتبع الفرع (D1) وإذا تعددت العناصر فاننا نتبع الفرع (D2).
- ٣ - إذا كان العنصر فى القائمتين متماثلاً (Identical) فإن ذلك ينهى العملية بنجاح وتكون النتيجة قائمة صفرية (NIL) .
- ٤ - إذا كان كل عنصر فى القائمتين معبراً عن ثابت متماثل فإن التوحيد يتم بنجاح ونحصل على القائمة الصفرية (NIL) وإذا لم يكونا متماثلين فإن العملية لا تكون ناجحة (FAIL).
- ٥ - إذا كان كل عنصر فى القائمتين معبراً عن نفس المتغير فإن عملية التوحيد تنتهى بنجاح ونحصل كذلك على القائمة الصفرية .
- ٦ - إذا كان العنصر فى القائمة الأولى متغيراً أو العنصر فى القائمة الثانية متغيراً فاننا نحصل على قائمة التعويض فى الشكل (List-b/List-a) أو (List-a/List-b) .
- ٧ - يمثل الفرع (D2) عملية التوحيد بين قائمتين متعددتين العناصر حيث يجرى فصل العنصر الأيسر من كل قائمة على أنه رأس لها أى (Head-1, Head-2) ثم يجرى تطبيق الجزء الأيمن (D1) من خريطة التدفق عليها ، وتصبح العملية معقدة حيث يتم استخدام نتيجة التعويض فى خطوة ما فى الخطوة التى تليها.
- ٨ - يتم كتابة المتبقى من القوائم (Rest-1, Rest-2) وذلك بإحلال المتغيرات مع قيم التعويض السابق الحصول عليها فى الخطوة السابقة كما يلى :
- إذا فرضنا أن (M1) هى قائمة التعويضات فى الخطوة السابقة فإنه يجرى استخدامها فى التعويض فى المتبقى من القوائم كما يلى :

Apply M1 to Rest-1

Apply M1 to Rest-2

$M2 = U1(Rest-1, Rest-2)$.

مثال : لإجراء عملية التوحيد لقائمتين باستخدام (U1) .

نفرض ان هناك قائمتان تأخذان شكل المعادلات التامة التركيب (Wffs) فى الصورة

الآتية :

List-a = (question What is)

List-b = (question has What)

من الملاحظ أن كلمة المتغير (What) تظهر فى كل من القائمتين وتجرى عملية التوحيد

كما يلى :

١ - يتم فصل العنصر الأول من كل قائمة كرأس لها ، فى هذه الحالة فإن الرأس فى الحالتين

واحدة وهى كلمة (question) ويعقد الربط بين الرأسين وتصبح قائمة التعويض قائمة

صفريّة (NIL) وهى أو ناتج لعملية التوحيد .

٢ - عند إجراء التعويض بالقائمة الصفريّة مع المتبقى من القائمتين فإن ذلك لا يغير من الامر شيئاً ويظل المتبقى كما هو.

٣ - يصبح المتبقى من القائمتين كما يلى :

Rest-1 = (What is)

Rest-2 = (has what)

ويتطابق (U1) على كل من المتبقى للقائمتين فإننا نحصل على قائمة التعويض :

M1 = (has / What)

والتي تعنى أنه فى هذه الخطوة قد تم ربط قيمة المتغير (What) بالقيمة (has) .

٤ - وعند تطبيق (M1) على المتبقى من القائمتين تصبح كما يلى :

M1 (is)

M1 (What)

فان ذلك يعطى قائمة جديدة :

(is)

(has)

وتصبح عملية التوحيد غير ناجحة (FAIL) حيث أن قيمة الثابت للمتغير (What) ليس لها قيمة واحدة فى الحالتين.

(٨-١١-٤) التوحيد بمقارنة قوائم الربط

(Unification by Checking the Binding Lists)

يمكن استخدام مقارنة قوائم التعويضات أو الربط لإجراء عملية التوحيد وفيما يلى الخوارزم

الثانى (U2) الذى تم كتابته باستخدام تعبيرات لغة باسكال :

PROCEDURE unify2(List-a,List-b);

BEGIN

IF List-a or List-b consist of a single element (an atom)

THEN

IF List-a and List-b are identical { indicates same variables case}

THEN return empty list , NIL {no substitutions}

ELSE

IF List-a or List-b are variables { logical OR}

THEN sequentially check the following cases and take the corresponding action ;

CASE:

1.List-a unbound variable : return (List-b / List-a)

2.List-b undound variable : return (List-a/ List-b)

{Otherwise List-a and List-b must either be bound or not variables}

3.(List-a or binding on List-a)consistent with (List-b or bind-

```

        ing on List-b):return NIL
    4.return FAIL
    ELSE return FAIL {not identical and neither is a variable }
    ELSE {not atoms ; must be lists}
    BEGIN {decimate input lists}
        Head1:=1st element of List-a;
        Rest1:=rest of List-a;
        Head2:=1st element of List-b;
        Rest2:=rest of List-b;
        M1:=unify2 (Head1,Head2);
        IF M1=FAIL
        THEN return FAIL
        ELSE
        BEGIN
            M2:=_unify2 (Rest1,Rest2);
            IF M2 = FAIL
            THEN return FAIL
            ELSE form new binding list as composite of M1 and M
                and return it
        END;
    END;
END ;{UNIFY}

```

والذى يوضح مقارنة قائمة التعويض بعد كل خطوة .مثال : إذا فرضنا القائمتين الآتيتين :

P1 = (block_1 left block_2)

P2 = (A left_of B)

فإن قائمة التعويض اللازمة للتوحيد بين القائمتين هي :

S={(block_1 / A),(blick_2 /B)}

الفصل التاسع

**نظم الاستدلال
المعتمدة على القواعد**

**Rule - Based
Inference Systems**

(١-٩) نظم الإنتاج (Production Systems)

تعتبر نظم الإنتاج أحد النظم الشائعة الاستخدام لتمثيل ومعالجة المعلومات، حيث يمكن الوصول إلى الحلول للعديد من تطبيقات الذكاء الاصطناعي باستخدام المعالجة الرمزية، وتعتبر هذه الأنظمة بسيطة المفهوم ويمكن تجهيزها في شكل برامج جاهزة أو مولدات للتطبيقات لتتبع الاستخدام بدون الحاجة إلى مبرمجين أو متخصصين في التطبيقات المختلفة للذكاء الاصطناعي. مثال لذلك قاعدة الإنتاج (OPS5).

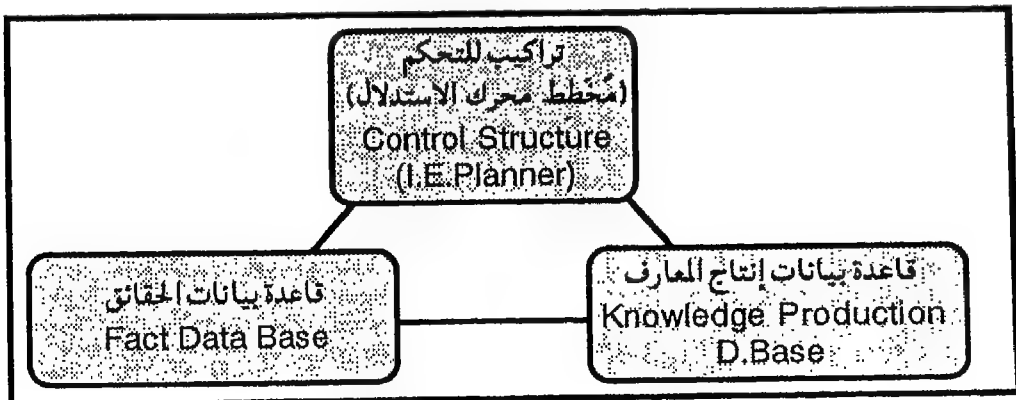
يبين شكل (٩-١) مثلث نظام قواعد الإنتاج، حيث يتكون النظام مما يلي :

١ - قاعدة بيانات الحقائق (Facts Data Base)

٢ - قاعدة بيانات إنتاج المعارف (Knowledge Production D.B.) والتي تكون مجموعة من النواتج (Productions) في شكل قواعد (Rules) والتي تعمل علي تطوير قاعدة البيانات الأصلية حيث تصبح قابلية تنفيذ هذه القواعد مشروطة بوجود الحقائق في قاعدة البيانات الأصلية.

٣ - تراكيب للتحكم والتي يطلق عليها ميكانيكية تحكم (Control Mechanism) ومترجم ومفسر للقواعد (Rule Interpreter) الذي يحدد قابلية تنفيذ القواعد في قاعدة البيانات الجارية واختيار القواعد المناسبة وتحليل التضارب الذي قد يظهر عند تنفيذ إنتاجين أو أكثر في نفس الوقت.

ان الإستنتاج لقاعدة ما في نظام قواعد الإنتاج يتطلب توصيف مجموعات ثنائية من التضمين الشرطي (IF- THEN) الذي يتحقق من خلال الحقائق المدخلة التي يمكن أن تأخذ شكل القوائم كما في لغة ليسب حيث يتم اجراء التوافق بينهما، وبذلك يتبين اعتماد نظم الإنتاج



شكل (٩-١) مثلث قواعد الإنتاج

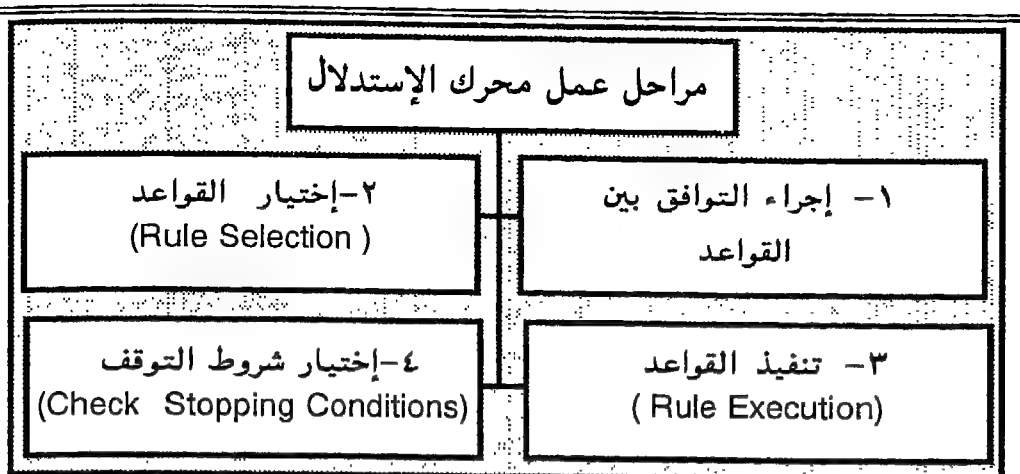
بالدرجة الأولى على التضمين الشرطى الذى يؤدى بدوره إلى إستنتاج قواعد جديدة من الحقائق المعرفة فى مجال ما ، ومن هنا جاء تعريف النظم الخبيرة (Expert Sytems) على أنها جزء من قواعد الإنتاج فى مجال معين .

كما سبق يمكن تعريف لغة البرمجة المنطقية برولوج مثلاً على أنها نظام مبنى على القواعد (Rule Based System) التى تعمل من خلال الحقائق والقواعد للوصول إلى احراز هدف ما ، علاوة على كونها لغة برمجة مبنية على الإسلوب المنطقى.

(١-١-٩) محرك الإستدلال أو الإستنباط (Inference Engine)

يعتبر محرك الإستدلال وقاعدة البيانات للحقائق (Fact D.B) قلب نظام الإنتاج المعتمد على القواعد (RBS) ، حيث تعتبر قاعدة البيانات هى ميدان العمل لمحرك الإستدلال ، وإذا تم إعتبار محرك الإستدلال كآلة ذات مراحل عمل معينة منتهية ، فإن مراحل العمل لهذه الآلة كما فى شكل (٩-٢) تشمل مايلى :

- ١ - إجراء التوافق بين القواعد (Rule Matching) .
 - ٢ - إختيار القواعد (Rule Selection) .
 - ٣ - تنفيذ القواعد (Rule Execution) .
 - ٤ - إختيار شروط التوقف عند إحراز الهدف (Check Stopping Conditions)
- بذلك يمكن إعتبار مراحل عمل محرك الإستدلال على أنها عملية إنضباطية (itrative) تتحكم فى استنتاج قواعد جديدة كما فى التسلسل الامامى (Forward Chaining) أو التحقق من المعلومات الافتراضية كما فى التسلسل الخلفى (Backward Chaining) .
- تعتبر عملية إجراء التوافق هى الأساس فى بناء محرك الإستدلال حيث يجرى تشفير (coding) المعلومات الأكثر عمومية بقواعد معرفة بإستخدام المتغيرات وميكانيكية ملائمة للتوحيد. كما أنه يمكن إعتبار عملية إختيار القواعد عملية حرجة فى الأداء لمحرك الإستدلال لإشتمالها على مايلى :
- ١ - إختيار القواعد المناسبة والقابلة للتطبيق.
 - ٢ - إختيار أنسب القواعد التى تنتج الحل المناسب للمشكلة قيد الحل بدون إستخدام طرق البحث المضنية أو إستخدام نظرية الاحتمالات.
 - ٣ - تحليل حالات القواعد التى تسبب تضارب وتضاد وإختيار المناسب منها.



شكل (٩-٢) مراحل عمل محرك الإستدلال

(٩-١-٢) السمات العامة لنظم الانتاج (Features of Production Systems)

تعتبر الخواص الآتية من السمات العامة لنظم الانتاج المعتمد على القواعد:

١- سهولة تحديث القواعد والحقائق مثل الاضافة والحذف فى قاعدة البيانات.

٢- سهولة الوصول الى قاعدة المعرفة فى شكل قابل للقراءة.

٣- المرونة فى المعالجة.

٤- سهولة تتبع عمل ميكانيكية الاستدلال.

٥- الإستخدام القياسى لتمثيل المعلومات وكذلك عملية الاستدلال.

٦- قابلية إستخدام البرامج التطبيقية الجاهزة.

ومن السلبيات لهذه النظم ما يلى:

١- صعوبة التنبؤ بسلوك النظام بشكل محدد.

٢- من الصعوبة لدفع ترتيب تتابعى إنتاجى معين وذلك على خلاف لغات الأمر.

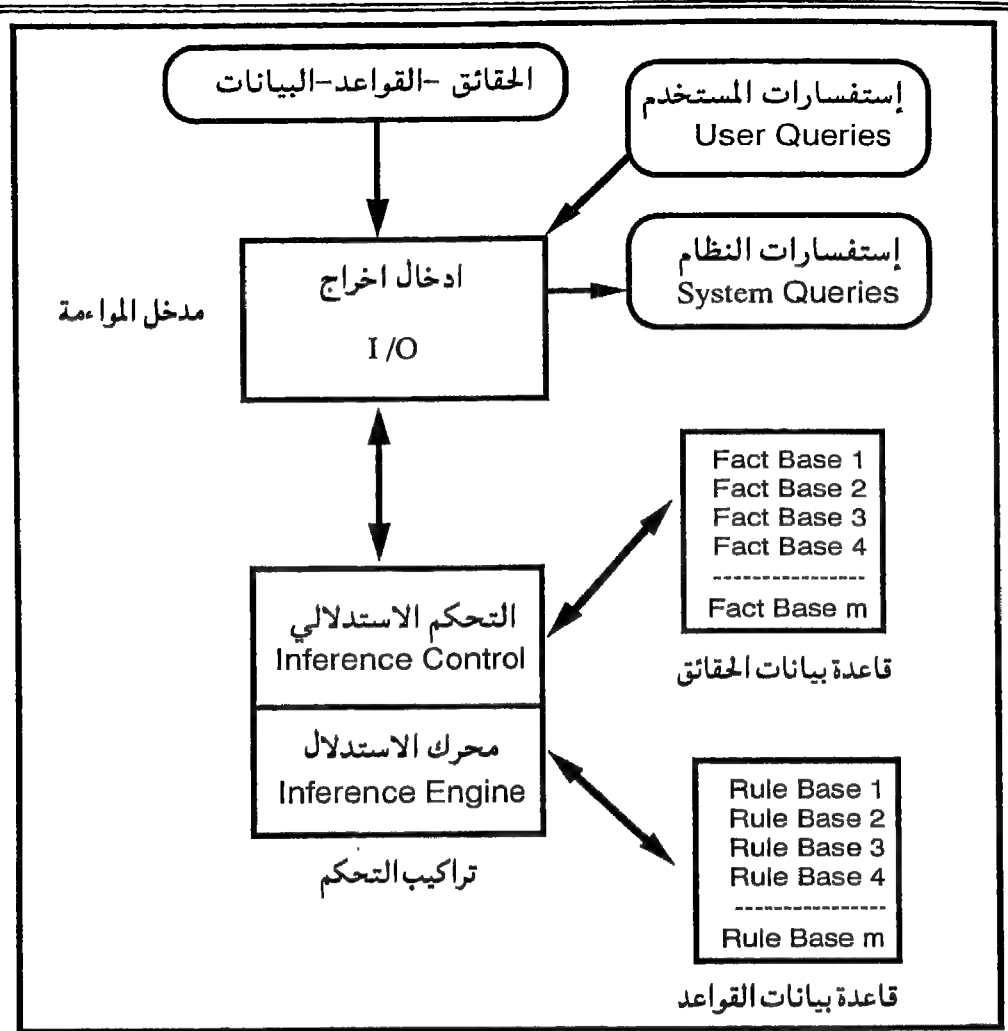
٣- عدم قابلية التطوير لكثير من التطبيقات.

وبين شكل (٩-٣) معمارية نظام إنتاج معتمد على القواعد والذي يمثل نظام للخبرة.

(٩-١-٣) محرك الإستدلال وإيجاد الحل للمشكلات

(Inference Engine & Problem Solving)

عند تجهيز نظام للانتاج من خلال إستخدام التسلسل الامامى او التسلسل الخلفى او كلاهما معاً فإن شبكة الاستدلال الناتجة تشرح ان محرك الاستدلال سوف يقوم باستخدام مسار او اكثر لمجموعة الحقائق لتحقيق هدف ما ، كما يحدث هذا ايضا فى حالة التسلسل الخلفى ، وبذلك تظهر



شكل (٩-٣) معمارية نظام إنتاج معتمد علي القواعد

اعداد كبيرة من المسارات الزائدة او الغير منتجة، لذلك يجب ان يراعى عند تصميم هذه الآلة ان تكون كل المعلومات ذات الخصائص التبادلية والتحليلية متاحة للتطبيق لتجنب حدوث ذلك.

(٩-١-٤) خصائص نظم الانتاج (Production System Properties)

يعتبر المفهوم النظرى والتطبيقاتى لنظم الانتاج من الوضوح بحيث ان قاعدة البيانات ومجموعة القواعد المستخدمة هي التي تحدد الشكل العام لمحرك الاستدلال وميكانيكية الاستنباط المحتمل إستخدامها، كما ان كثير من الخصائص تعود الى التطبيق الأمثل للقوانين والقواعد المستخدمة. وفيمايلي بعض الخصائص :

١ - القواعد القابلة للتنفيذ والنظم الإبدالية :

تعتبر النظم الابدالية من النظم المتخصصة ويمكن القاء الضوء على هذه النظم كمايلي:
إذا قمنا بتمثيل القاعدة القابلة للتنفيذ بالحرف (D) فى قاعدة البيانات فإن نظام الانتاج
الإبدالى يتميز بالخواص الآتية:

أ - تعتبر أى قاعدة قابلة للتنفيذ على (D) قابلة للتنفيذ على أى قاعدة بيانات مشتقة
من (D)، كما يحتمل انتاج قواعد اخرى تكون لها نفس قابلية التطبيق.

ب - إذا تحقق الهدف بتطبيق القاعدة على (D) فإنه يتحقق كذلك على أى منتج لقاعدة
البيانات بتطبيق القواعد المنتجة والقابلة للتنفيذ على (D).

ج - تأخذ قاعدة البيانات المتولدة من التشغيل المتتالى للقوانين القابلة للتنفيذ على (D)
نفس الشكل التبادلى فى التشغيل المشتقة او الوسيطة التى تصل الى احراز الهدف.

تتضح اهمية وجود نظام إنتاج مع خاصية ابدالية فى المثال الآتى :
نفرض ان هناك قاعدة بيانات تحتوى على عدد (٥) حقائق وعدد (٦) قواعد والمثلة فى
الخطوة (١).

٢ - للوصول الى تحقيق الهدف (h) الذى يتمثل فى القواعد (الاولى والثانية) فإننا سوف نقوم
بعرض اربع شبكات إستدلالية ابدالية لتحقيق نفس الهدف والمبينة فى الخطوة (٢).
ومن هنا نرى انه طبقاً لخاصية الابدال فإن الترتيب يكون غير ضرورى، وتتضح الاهمية
الازدواجية كمايلي:

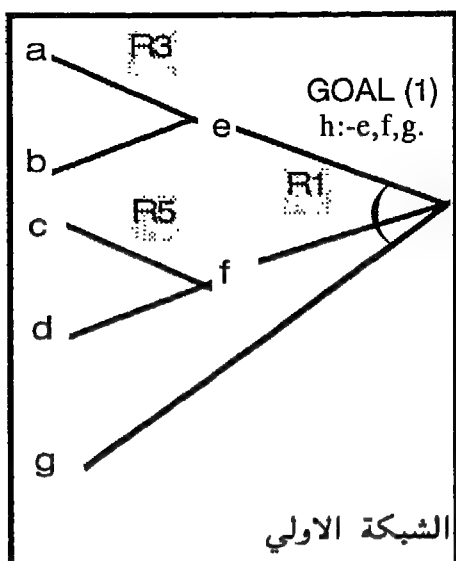
١ - لايؤخذ فى الاعتبار تطبيق كل تبديلات ترتيب القوانين عند تصميم محرك الاستنباط،
وبذلك يمكن تجنب كثير من مسارات الحلول التى تختلف فقط فى الترتيب خصوصاً اذا كانت
القواعد المطبقة كبيرة الحجم .

٢ - تتيح خاصية الابدال استخدام محرك الاستدلال بأشكال مختلفة تؤدي نفس الغرض.

(٩-١-٥) نظم الانتاج والتخطيط (Production System and Planning)

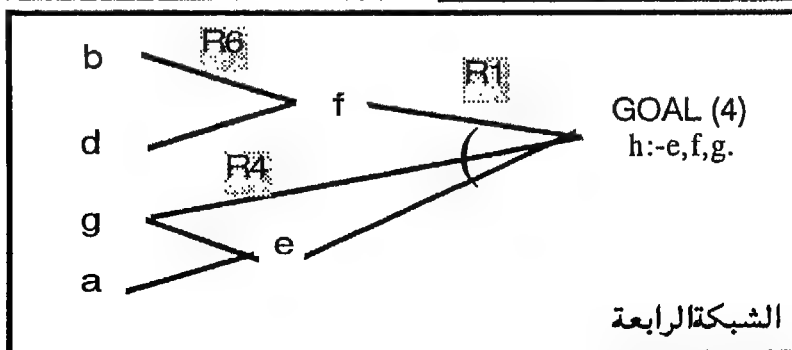
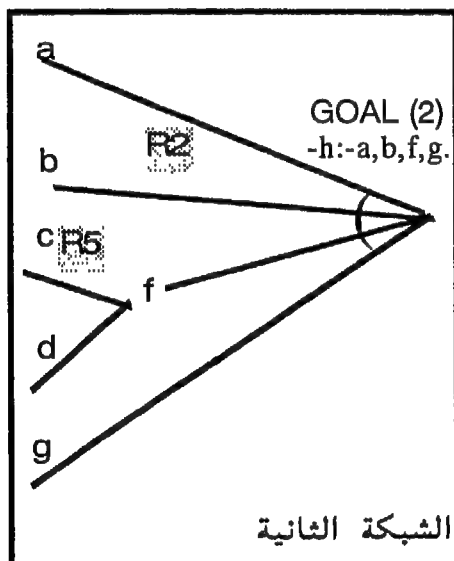
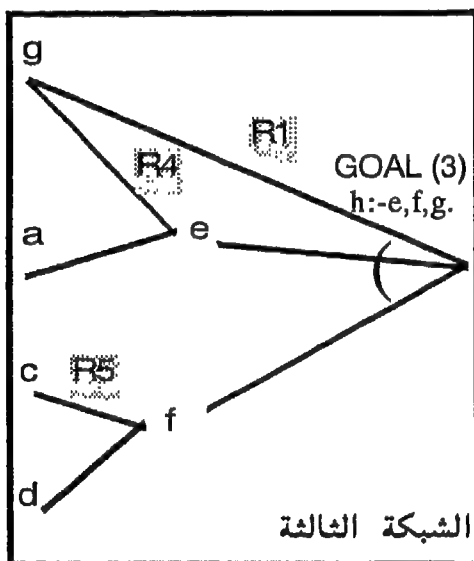
تعتبر نظم الانتاج الغير إبدالية والتى تتميز بالمقدرة على حذف الجقائق والقواعد والتى
تماثل لغة برولوج فى استخدام الدالة الاسنادية للحذف (Retract Predicate) من الأمثلة
الناجحة فى تطبيقات الذكاء الاصطناعى خصوصاً فى برامج الذكاء الاصطناعى للتخطيط
(Planning) حيث يكون ترتيب استخراج النتائج عادة فى وضع حرج، ومن جهة أخرى فإن
إنجاز الخطة التى تفي بمتطلبات الهدف الموصف تعتمد الى حد كبير على عملية إختيار تتابع
عمليات شمول الخطة الموضوعية.

٢- شبكات الاستدلال



١- قواعد البيانات

الحقائق	القواعد
Facts (D)	Rules (R)
a.	1-h:-e,f,g.
b.	2-h:-a,b,f,g
c.	3-e:-a,b.
d.	4-e:-g,a.
g.	5-f:-c,d.
	6-f:-b,d.



(١-١-٦) نظم الانتاج القابلة للتجزئة

(Decomposable Production System)

رغم أن الخاصية الابدالية لنظم الانتاج تسمح بدرجة محدودة من المرونة فى التتابع المشتمل على القوانين القابلة للتنفيذ، الا ان خاصية التجزئة تسمح ببعض الحرية الاضافية فى ترتيب تنفيذ القوانين، حيث تؤدي الى كفاءة عالية فى حسابات تصميم محرك الاستدلال، حيث تعتبر التجزئة لمشكلة ما الى أجزاء أصغر ذات حلول منفصلة عاملاً مساعداً للوصول بسرعة الى الحل الشامل. وتعتبر قاعدة البيانات (D) قابلة للتجزئة لو أمكن تحليلها الى فئات غير مترابطة يمكن معالجتها باستقلالية، علاوة على ان يكون الهدف أيضاً قابل للتجزئة الى مكونات متحققة من خلال قاعدة البيانات. ومن الأفضل ان يكون الهدف قابل للتحقيق باستخدام بعض الدوال المنطقية التي تربط بين نواتج العمليات المجرى مثل اداة الربط المنطقى (AND).

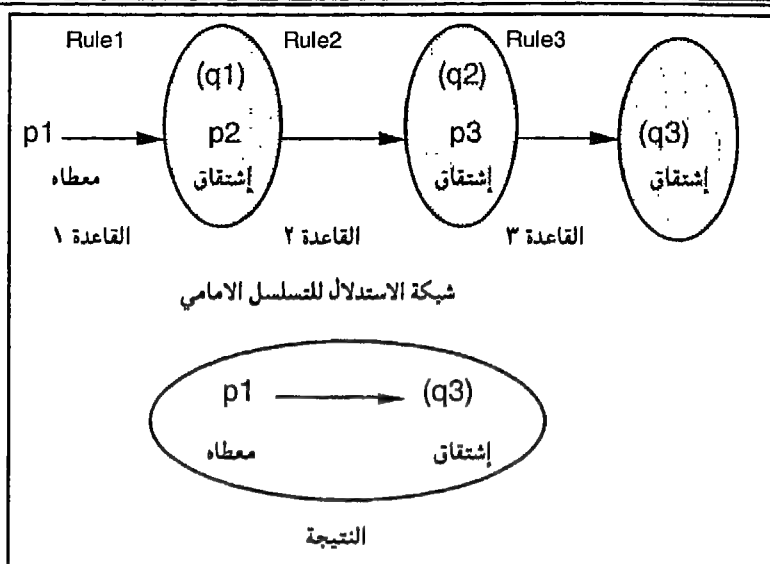
- كما هو واضح من الشبكات المبينة في المثال السابق حيث جرى تجزئة تنفيذ الهدف الى اجزاء منفصلة ، بذلك تعتبر خاصية القابلية للتجزئة هامة لسببين رئيسيين:
- ١ - تسمح التجزئة بالحصول على استدلالات متوازية لمدخلات تحليل الهدف.
 - ٢ - تسمح التجزئة بحذف كثير من المسارات للحلول الزائدة.

(٢-٩) الاستدلال المتسلسل المعتمد على القواعد

(Rule-Based Chaining Inference)

- كما ذكرنا سابقاً من ان عمل محرك الاستدلال يتلخص فيما يلى:
- ١ - اختيار القواعد او فئاتها المناسبة بسيناريو معين تحدده ميكانيكية التحكم.
 - ٢ - التوافق او المواءمة التي تقوم بتحديد ملاءمة القاعدة المعطاه مع النص بقاعدة البيانات.
 - ٣ - ترتيب إطلاق القواعد وتحليل التضارب الذي ينشأ من إنتاج قواعد جديدة ربما تتناقض مع المعلومات المتاحة الجارية.
 - ٤ - تنفيذ القواعد بإطلاقها وبناءاً عليه يتم التطوير الجزئى لقاعدة البيانات .
 - ٥ - التحقق من تحقق الهدف الكلى.
- وكما تم عرضه سابقاً عن التسلسل الامامي فإننا نعيد عرضه بالشكل الملخص الآتى:
- ١ - نفرض قاعدة القواعد الابتدائية والتي تتكون من جمل التضمنين الشرطى التالية:

$p1 \Rightarrow p2$



p2 → p3

p3 → q3

٢ - تصبح شبكة الاستدلال كما بالشكل.

٣ - من الواضح في شبكة الاستدلال المبينة كيف أمكن استخدام الاشتقاق والتسلسل الامامي للوصول الى النتيجة، ويوضح الشكل المنقط حيز اطلاق القاعدة، ومن الواضح ان تطبيق تركيبات التحكم بالنسبة للتسلسل الامامي يكون مباشراً ويكن القول بان عملية تحديد القاعدة الواجبة للاطلاق، والتي يطلق عليها عملية تحليل التعارض (Conflict Resolution) تبدأ بمسح قاعدة البيانات للقواعد بهدف ايجاد القاعدة المناسبة لتحقيق الهدف وذلك لاطلاقها.

(٩-٢-١) إعداد نظام تسلسل أمامي بإستخدام الـ ليسب

(Forward Chaining System using LISP)

يتركب نظام الانتاج المبنى على القواعد كما ذكرنا سابقاً من الثلاث وحدات الآتية:

- ١ - مجموعة من الحقائق الاساسية (Facts) التي يجرى تشفيرها كقائمة (List) .
- ٢ - مجموعة من القواعد فى شكل صيغة التضمين الشرطى (IF-THEN) والتي يجرى تشفيرها كقائمة (List) .
- ٣ - تطبيق مجموعة القواعد الشرطية (البند ٢) تتابعياً على مجموعة الحقائق الأساسية (البند ١)، وذلك من خلال عمل شبكة الاستدلال التى يمكن أن ينتج عنها حقائق جديدة او

إستنتاجات ذات مستوى عال.

من الواضح أن توليد حقائق جديدة يستمر عندما تتداخل أجزاء النتيجة التابعة للجزء (THEN) مع الجزء الشرطى (IF) للحقيقة التى تليها، وتستمر حتى يتم الوصول الى الهدف او حتى لا يكون هناك إنتاج لحقائق جديدة.

يبين شكل (٩-٤) الاستراتيجية العامة لبناء التسلسل الأمامى التى تحتوى على مفهوم تحليل التعارض الذى يشكل جزءاً أساسياً فى بناء النظام ككل.

(٩-٣) التسلسل الخلفى (Backward Chaining)

من المعروف انه عند تمثيل النظم من المراحل الابتدائية الى المرحلة النهائية او المرحلة الهدفية ان يتناوب التسلسل الخلفى مع التسلسل الامامى، فبينما نجد ان التسلسل الامامى يبدأ دائماً بقاعدة البيانات الابتدائية ويسير فى إتجاه مرحلة الهدف متبعاً بذلك مسار او اكثر، فان التسلسل الخلفى او التسلسل ذو التفرع المتتالى هو معكوس التسلسل الامامى وان هذا الانعكاس ليس انعكاساً بسيطاً للتضمن الشرطى.

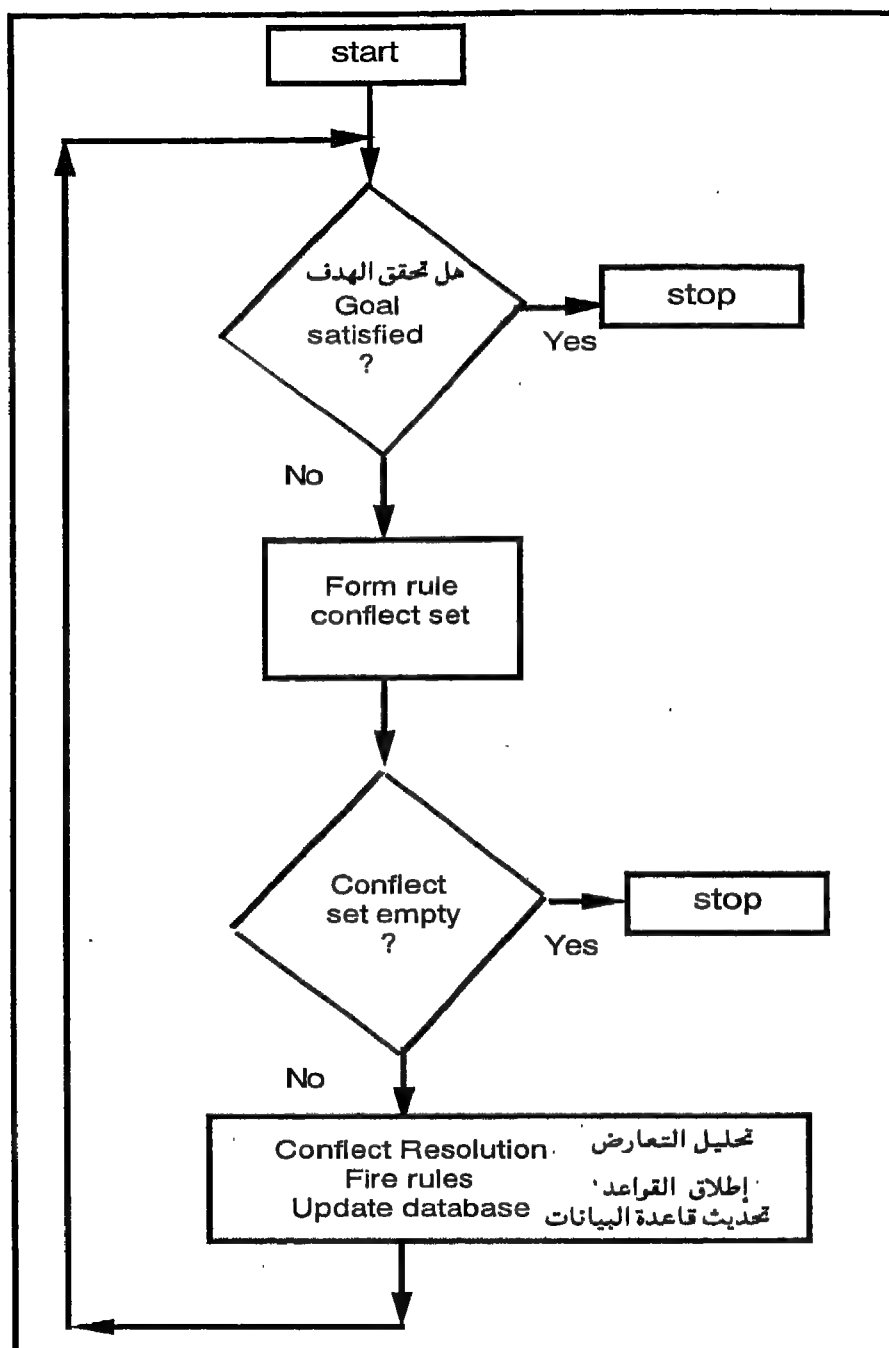
يمكن القول بان التسلسل الخلفى هو إستخدام التوابع او النتيجة المرتبطة بالجزء الثانى من التضمن الشرطى (THEN) لتوجيه البحث لايجاد قواعد للاطلاق. وكما هو موضح فى شكل (٩-٥) فإن العمل او النتيجة (b) سوف تستخدم لايجاد الشرط او الحدث (a) الذى يتسلسل خلفياً الى اطلاق القاعدة الثانية من خلال النتيجة (d) ثم الشرط (c) ثم متابعة التسلسل الى اطلاق القاعدة الثالثة (f) ثم (e).

(٩-٣-١) أساسيات التسلسل الخلفى (Basis for Backward Chaining)

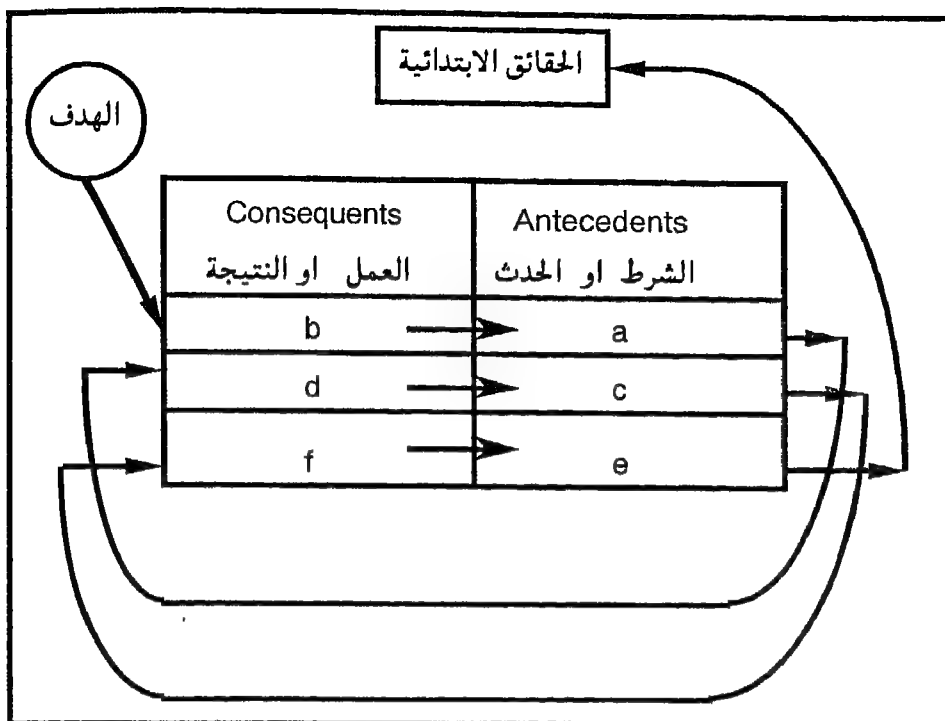
لتصميم وفهم ميكانيكية التسلسل الخلفى فإننا نتبع نظام مكون من ثلاث قواعد، ويبين شكل (٩-٦) خطوات شبكات الاستدلال للتسلسل الخلفى، والذى يبدأ بفرض أن الهدف الأساسى (q3) صحيحاً، ومن خلال إطلاق القاعدة (٣) فإن الهدف الثانى (2nd sub goal) سوف يتحقق فى شكل الشرط (p3)، ثم يتسلسل خلفياً للوصول الى الهدف الثالث (3 rd sub goal) (p2)، وذلك بإطلاق القاعدة (٢) والذى يكون صحيحاً ثم يتسلسل لاطلاق القاعدة (١) للوصول الى الحقيقة المعطاة (p1).

من الواضح ان التسلسل الخلفى هو فى الحقيقة عكس قاعدة التضمن الشرطى فى الصورة

$$q \rightarrow p$$



شكل (٩-٤) خريطة تدفق لتحليل التعارض وتحديث قاعدة البيانات



شكل (٩-٥) استخدام التوابيع لتوجيه البحث لاطلاق القواعد

ويمكن شرح مثال آخر يتناول القاعدة الآتية :

(rule r1 (if f c g) (THEN z))

عند توصيف الهدف (z) يلزم توصيف عدد من الهديفات (subgoals) الاتية
(f c g)

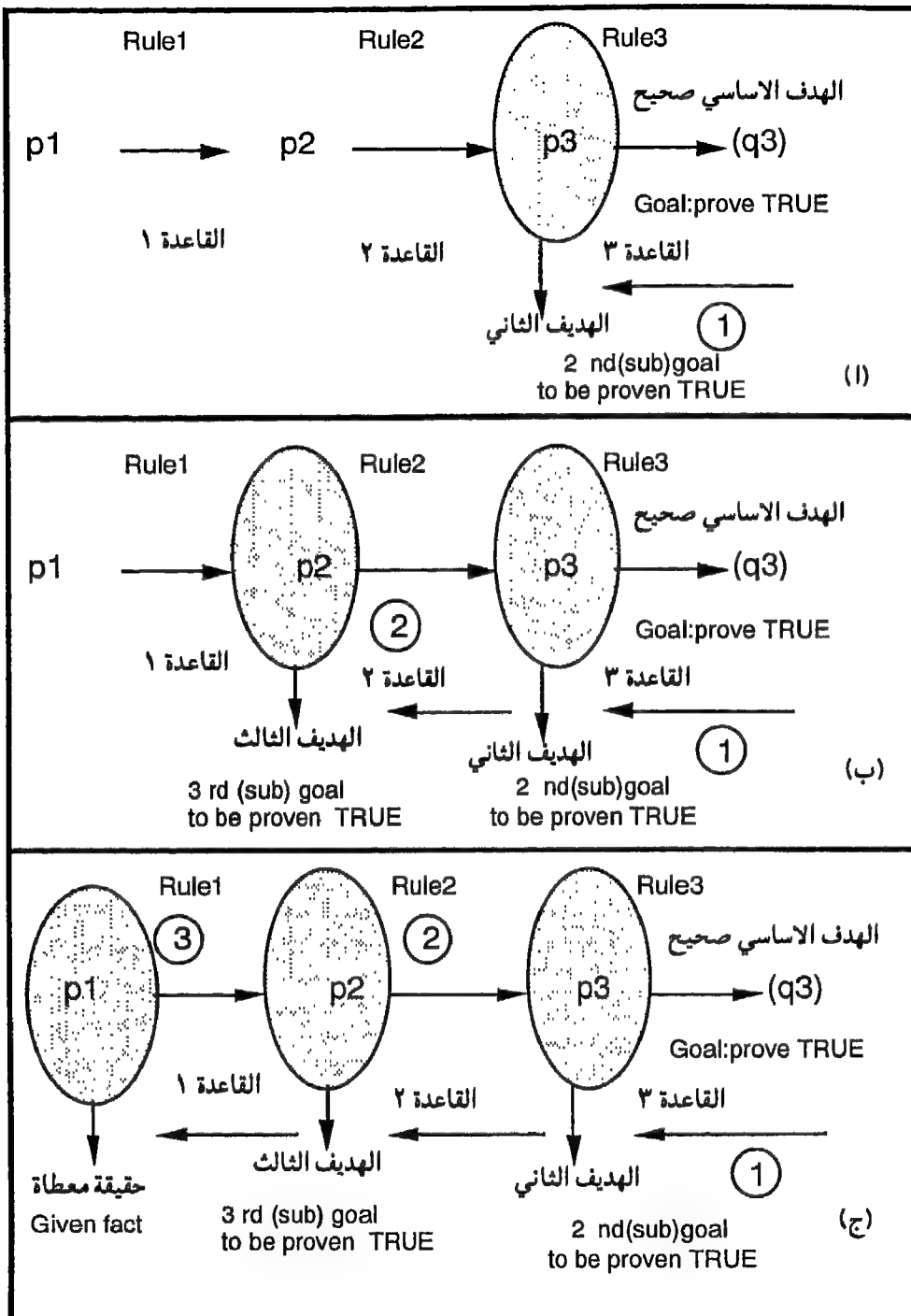
وتصبح قائمة القواعد :

```
(setq *rules* '(
(rule1 (IF f c g) THEN z))
(rule2 (IF d e) THEN f))
(rule3 (IF a b) THEN c)))
```

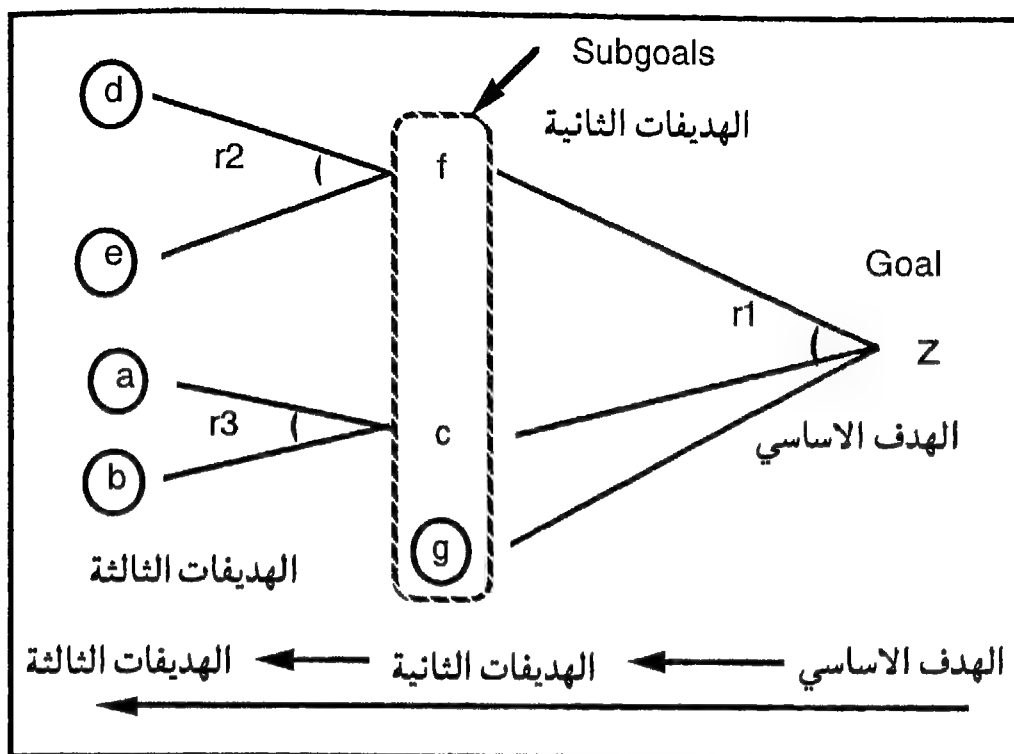
وكذلك قائمة الحقائق:

```
(setq *facts* '(a b d e g))
```

وبذلك يكون تحقيق الهدف (z) باستخدام التسلسل الخلفي ملازما لتحقيق الهديفات الثانية
(f c g) والهديفات الثالثة (d e a b) كما هو موضح في شكل (٩-٧).



شكل (٩-٦) خطوات شبكات الاستدلال لتسلسل الخلفى



شكل (٧-٩) تحقيق الهدافات للوصول الى الهدف الاصلى

الجزء الثالث

لغات البرمجة والتطبيقات

Programming Languages

and Applications

الفصل العاشر

الذكاء الاصطناعي

ولغات البرمجة

AI and

Programming Languages

(١-١٠) المعالجة الرمزية ولغات البرمجة

(Symbolic Processing & Programming Languages)

من المعروف أنه لا توجد لغة برمجة صالحة للتطبيق ١٠٠٪، ولكن يمكن القول بأن اللغة المناسبة هي اللغة التي تسهل للمستخدم إجراء التوصيف (Description) وإجراء العمليات الحسابية (Computations) للمشكلة أو التطبيق قيد الدراسة، هذا بالإضافة إلى الكفاءة والقدرة على الانتشار. بينما تكون لغة كوبول (COBOL) مناسبة للتطبيقات التجارية، ويتعلم المبتدئون لغة البيسك (BASIC)، وتستخدم لغة فورتران (FORTRAN) للتطبيقات العملية، فإن لغتي البرمجة المنطقية برولوج (PROLOG) ولغة البرمجة بإسلوب القائمة (LISP) تناسبان الاستخدام في تطبيقات الذكاء الاصطناعي لتمييزهما الواضح في المعالجة الرمزية (Symbolic Processing). مع ملاحظة أنه يمكن استخدام اللغات التقليدية في بناء نظاماً للذكاء الاصطناعي الآن هذه اللغات تكون من الصعوبة في التنفيذ بالمقارنة باللغات الرمزية.

(١-١-١٠) المعالجة الرمزية والمعالجة التقليدية

(Symbolic and " Conventional " Processing)

تعتمد برامج الذكاء الاصطناعي بالدرجة الأولى على التراكيب الرمزية والشكلية (Symbolic) والعلاقات بينهما ولا تعتمد على ال (Bits) وال (Bytes) ولا الدقة في تحديد الأرقام (Precision)، وبين الجدول المبين في شكل (١-١٠) الفرق بين المعالجة الرقمية التقليدية والمعالجة الرمزية.

(١٠-٢) نظم المعالجة المختلطة (Coupled Processing Systems)

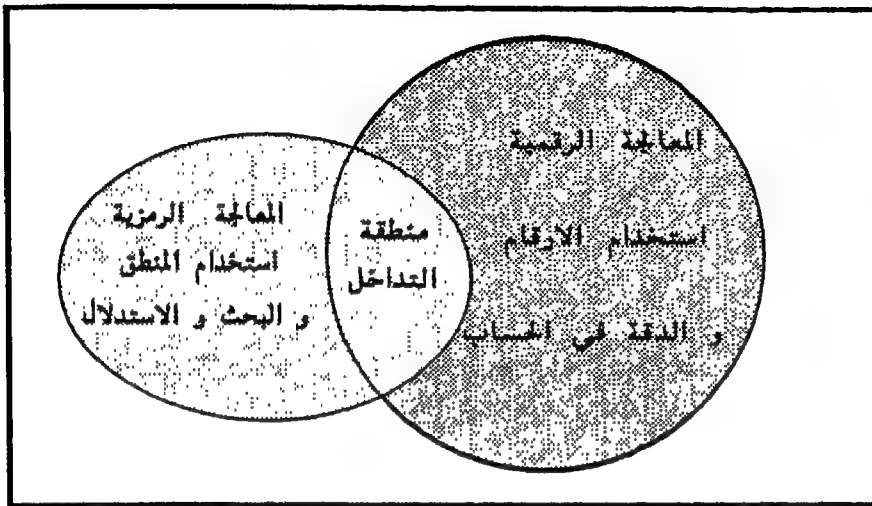
في كثير من التطبيقات العملية للذكاء الاصطناعي يلزم استخدام المعالجة الرمزية جنباً إلى جنب مع المعالجة الرقمية، حيث يمكن أن تكون البداية في تطبيق ما استخدام المعالجة الرقمية التي تستخرج بيانات تعالج رمزياً بعد ذلك مثال ذلك : عند معالجة الصور (Image Processing) فإنه يلزم استخدام المعالجة الرمزية مثل خوارزميات التوحيد التي تتطلب أولاً استخراج ملامح الصورة باستخدام معالجة رقمية منخفضة المستوى ثم معالجة رمزية بعد ذلك، ومن هنا جاء تعريف النظم المختلطة للمعالجة وكما هو مبين في شكل (١٠-٢) حيث تعتبر منطقة التداخل بينهما هي المنطقة التي تختلط فيها المعالجة الرقمية والمعالجة الرمزية.

تعتمد معالجة نظم الذكاء الاصطناعي بالدرجة الأولى على لغات البرمجة المنطقية (Logic Prog.) ولغات البرمجة الدالية (Functional Prog.) كما يمكن توضيح استخدام البرمجة

المعالجة الرمزية Symbolic Processing	المعالجة الرقمية Numerical Processing
١- تقوم المعالجة علي الرموز والفروض والبدهييات والقواعد والعلاقات .	١- تقوم المعالجة على الأرقام والدوال والايخارج النمط (Formatted Output) اعتماداً على ال (Bits) ، و (Bytes) .
٢- تستخدم لغة وصفية (Descriptive) لشرح الحقائق والعلاقات .	٢- تستخدم لبناء البرامج خوارزميات إجرائية (Procedual) لحل المشكلات .
٣ - يتم الإستعانة بطرق البحث مثل البحث الهرمي (Heuristics Search) لإيجاد الحلول .	٣- يتم الإستعانة بترتيب متابعي معين لتنفيذ خطوات الحل .
٤ - يتم الحصول على نتائج غير محددة اذا كانت الخوارزميات المستخدمة تميل الي التشتت .	٤- يتم الحصول على نتائج محددة . (Deterministic Result)
٥ - نوعية البيانات : ذرات وأشياء وكيانات وبدهييات وقوائم .	٥- نوعية البيانات : حروف وارقام .
٦ - المتغيرات غير معرفة ويلزم فى بعض الأحيان إنشاء متغيرات للمساعدة فى إيجاد الحل .	٦- المتغيرات تامة التعريف .
٧ - المتغيرات وتراكيب البيانات غير ثابتة التحديد حيث يمكن ان تزيد أو تقل.	٧- المتغيرات وتراكيب البيانات ثابتة التحديد.
٨ - عدم الدقة فى تمثيل المعلومات .	٨ - الدقة فى تمثيل المعلومات .
٩ - الإجابات المتوقعة من البرنامج غير محددة وأى نتيجة تعتبر مرضية.	٩ - الإجابات المتوقعة من البرنامج دقيقة التحديد

شكل (١٠-١) الفرق بين المعالجة الرقمية والمعالجة الرمزية

المنطقية (Prolog) والبرمجة بأسلوب القوائم (LISP) وعلاقتها باللغات الأخرى مثل لغة فورتران ولغة باسكال .



شكل (١٠-٢) نظم المعالجة المتداخلة أو المختلطة

(١٠-٣) التمثيل الحسابي وتراكيب البيانات

(Manipulable Representations & Data Structures)

من أهم أهداف البرمجة في نظم الذكاء الاصطناعي هو تمثيل الكيانات أو الأشياء (Entities & Objects) ووضعها بشكل مختصر (Abstract) يسهل التعامل معه حسابياً، فمثلاً تعتبر الأسماء مثل محمد وعلى وأحمد قيم رمزية (Symbolic Values) حيث توصف الحالات المختلفة المتغيرة (Instances) لكيان شخص (Person Intity) مثلاً ويختلف الأشخاص في السن ومستوى التعليم والعادات والصفات وغير ذلك. و لتوضيح الفرق بين لغة باسكال ولغة البرمجة بأسلوب القوائم ليسب فإننا سوف نقوم بتمثيل الكيان "شخص" كمايلي:

١- باستخدام لغة باسكال (Pascal) وذلك باستخدام أوامر التوصيف (TYPE), (VAR) كمايلي :

```
TYPE
person_type = record
    name      : string [ 14];
    age       : [0 ..100];
    education : set of(High School,College);
VAR person : ARRAY[1..N] of person_type ;
```

من الواضح فى لغة باسكال أنه تم إستخدام التمثيل الرمزى مثل الاسم (name) والتعليم (education) بجانب التمثيل الرقمى مثل العمر، وإستخدام بيانات عن الشخص "محمد" مثلا، فانه يمكن استخدام تراكيب لغة باسكال الآتية :

```
IF
  person.name[j]='Mohamed'
THEN
  mohameds-age := person.age[j]
```

٢- بإستخدام لغة البرمجة بأسلوب القائمة ليسب (LISP). يمكن التعبير عن ذلك ايضا باستخدام التراكيب المبنية داخليا فى لغة ليسب (built-in) كمايلى:

```
(putprop ' Mohamed' age 33)
and
(get ' Mohamed ' age )
```

أو باستخدام لغة ليسب العام كمايلى:

```
(set f(get ' Mohamed ' age 33).
and (get ' Mohamed'age).
```

من الواضح أنه عند إستخدام لغة ليسب حيث يتكون البرنامج من دوال (Functions) يجري تطبيقها علي البيانات (Data)، فان المبرمج لا يحاول ان يستنتج كيف يسير البرنامج ولكنه يركز علي مقدار تطبيق الدوال علي البيانات ومقدار ما يتم الحصول عليه من ذلك التطبيق، وفي لغة البرمجة المنطقية فان المبرمج لا يركز علي كيفية ترتيب خطوات التنفيذ، ولكنه يركز علي توصيف المشكلة وذلك بتطوير قاعدة البيانات المحتوية علي الحقائق والقواعد (Facts& Rules) حيث يتم الوصول الي الحل بإستخدام ميكانيكية التوحيد (Unification Mechanism) للإستدلال والإشتقاق.

(١٠-٤) الفرض الوصفى للبرمجة المنطقية

(Descriptive Concept for Prolog)

عند إنشاء وتطوير برامج بلغة البرمجة المنطقية يتبين الفروض التالية :

- عدم الضرورة الي الإستعانة بخرائط التدفق (Flow Charts) في أول الأمر حيث ان ذلك يتعارض مع الطبيعة الإعلانية (Declarative Nature) لهذه اللغة (جميع الحقائق والقواعد معرفة وموصفة) مع إستخدام ميكانيكية التوحيد المبينة داخل تراكيب المترجم أو المنفذ للغة (Interpreter or Compiler)، وبذلك يتكون البرنامج من الآتى:

١- الحقائق المعرفة (Known Facts).

٢- القواعد (Rules) التى تسمح بتجميع أو إشتقاق أو الأستدلال علي حقائق جديدة.

٣- الوصول إلى نتيجة البرنامج وذلك بإحراز الهدف (Goal) .
ومن الواضح أن تنفيذ خطوات البرنامج تقع كلية على المحلل أو المنفذ وذلك بتحديد وربط المتغيرات الموجودة بالحقائق بقيم يتم تحديدها بالخطوات التي تقترحها الموحدة (Unifires).

من أهم الاختلافات بين لغة البرمجة المنطقية ولغات البرمجة التقليدية مايلي:
تقوم البرمجة المنطقية علي توصيف للمشكلة قيد الحل (Problem Specification) والتي تقابل توصيف البرنامج (Program Specification) في اللغات الأخرى، حيث يشتمل توصيف المشكلة علي وضع البرنامج بدون إقتراح خوارزميات الحل، بينما يعتمد توصيف البرنامج على إقتراح خوارزميات الحل، وبذلك تعتبر لغة البرمجة المنطقية من أعلى اللغات (High Level) تركيباً، حيث أنها تسمح بإيجاد مسار يتوافق مع إشتقاق النتائج الصحيحة والتي تعكس مفهوم الإستدلال في الإنسان .

(١٠-٥) الشروط الواجبة في لغات الذكاء الاصطناعي

- يجب ان تتوافر الشروط الآتية في لغات الذكاء الاصطناعي:
- ١- القدرة علي تطوير نماذج و نظم ميكانيكية للتحليل والإثبات التي تتغير تغيراً دقيقاً، وذلك بتجزئة حل المشكلة الي أجزاء صغيرة ذات حلول مناسبة.
 - ٢- تركيب إنضباطي للتحكم المرن الذي :
 - (أ) - يساعد على توجيه خطوات البرنامج الى الحل السريع.
 - (ب) - يساعد على الوصول الي البيانات المناسبة لإعطاء الحل.
 - (ج) - يساعد على تسهيل التكرار المطلوب.
 - ٣ - تركيب إنضباطي للمساعدة في إجراء البرمجة المتوازية.
 - ٤ - قدرة النظام علي التحوار مع المستخدم.
 - ٥ - إستخدام برامج التصحيح في عملية مراجعة العمليات التكرارية وميكانيكية للتوحيد.
 - ٦ - بناء داخلي لإجراء تمثيل المعلومات رمزياً والوسائل التي تساعد علي إستحداث الأطارات لتمثيل المعلومات مثلاً.
 - ٧ - إستراتيجيات مختلفة لربط المتغيرات بقيم وتسهيل طرق البحث بإستخدام التجربة والخطأ لإيجاد الحل .

٨ - القدرة علي التوافق للبصمات (Pattern Matching) .

ومن المعروف أنه تم تصنيع وبناء شرائح الكترونية (chips) تعمل أساسا للتحليل والمعالجة الرمزية والمناسبة للغات الذكاء الاصطناعي (LISP, PROLOG) وذلك بإستخدام تقنيات الدوائر المتكاملة ذات السعات الكبيرة (VLSI) حيث أمكن بناء حاسبات الإستدلال القادرة علي إجراء المعالجة الرمزية .

(١٠-٦) المتطلبات البيئية لنظم البرمجة

(Environmental Requirements for AI Programming)

لبرمجة نظم الذكاء الاصطناعي لابد من تطوير البيئة لتشتمل علي نظم للموائمة مع المستخدم وتشتمل المتطلبات البيئية المناسبة على مايلي:

١ - إستخدام التطوير البنائي المتزايد (Incremental Program Development) كما هو واضح في لغتي البرمجة المنطقية والبرمجة بإسلوب القائمة، حيث يقوم البرنامج بتنفيذ قاعدة واحدة في الزمن الواحد حيث يتم اختبارها وتطويرها واستخدامها لبناء جزء اكبر من البرنامج، أى أن اللغة تساعد على تطوير وبناء نفسها كما فى لغة آدا (ADA) .

٢ - محرر متكامل وذكى (Integral and Intelligent Editor) يتطلب إستخدام محرر ذكى ومتكامل لكتابة وتصحيح شفرة المنبع (Source Code) حيث أن التشفير لهذه البرامج لا يتم دفعة واحدة ولكن يتم شفرة بعد الأخرى مما يساعد على تكامل التصحيح للبرنامج .

٣ - استخدام منفذ ومترجم تفصيلي (Incremental Compiler / Interpreter) حيث يعمل المترجم أولاً علي خطوات البرنامج خطوة بعد الأخرى تفصيلياً ثم يجرى تنفيذ خطوات البرنامج خطوة بعد الأخرى الى ان ينتهي البرنامج ككل. مثال ذلك فى لغة البرمجة المنطقية حيث يتم ادخال البنود أو حذفها من خلال الأوامر.

٤ - استخدام مصحح ذكى متوائم مع المستخدم (Intelligent and User-Friend Debuggers). تشتمل لغات البرمجة المنطقية والبرمجة بإسلوب القوائم علي إمكانية التتبع لخطوات البرنامج (Trace Facility) حيث يمكن قياس مستوي الاداء للبرنامج خصوصاً عند ربط المتغيرات وعملية التوحيد ووضع نقاط إيقاف عند تنفيذ البرنامج للتأكد من عمل البرنامج في الوجهة الصحيحة.

٥ - استخدام الامكانيات العالية الدقة والسرعة للعرض (High Speed and Resolution Display) حيث يمكن استخدام اسلوب الرسوم البيانية والشبكات

الدلالية والحصول علي نتائج بشكل رسومات مثل استخدام محطات العمل (Sun and Macintosh work Stations) وشبكات الحاسبات الشخصية.

٦ - استخدام المداخل والمخارج المتوائمة (Input and Output Interfaces) حيث يلزم في

بعض الأحيان قياس كميات من نظم حقيقية للإدخال أو استخدام الكميات عند المخرج في

بناء الشكل العام للنتائج المطلوبة، مثال ذلك عند التعرف ومعالجة نظم التحدث أو الصور

(Speech and Image Processing) .

٧ - استخدام ذاكرة تشغيل كبيرة الحجم (Large Memory Size) والتي اصبحت

متوافرة في الحاسبات الشخصية والتي تتلائم مع لغة برولوج ولغة ليسب، حيث يتم

إستخدام التكرار (Recursion) بشكل كبير.

(٧-١٠) تصنيف لغات البرمجة

(Programming Languages Classification)

يعتبر الفرض الخاص بتحديد أو تخصيص (Assignment) قيم للمتغيرات هو

الأساس في تقسيم اللغات العالية ، شكل (١٠-٣) ، والتي يمكن ان تنقسم الى مايلي:

١ - لغات الأمر (Imperative)

وهي اللغات التي يجرى تنظيمها حول التحديد أو التخصيص الفعلي للمتغير بقيم محددة

مستخدما في ذلك لغة الأمر مثل لغة فورتران (FORTRAN) حيث يقوم المبرمج أو

المستخدم بتحديد ترتيب الخطوات التي تؤدي الي الحل بإستخدام التحديد لهذا المتغير،

مثال ذلك... FOR i:=1 to n Do...

٢ - لغات التطبيق (Applicative)

وهي اللغات التي يجرى تنظيمها حول عمل أو تطبيق دالة (Function) علي

مدلول (Argument) وتعرف باللغات الإعلانية (Declarative) وذلك كمايلي:

أ - إذا إحتوت اللغة دوال ذات مدلولات تحدد بمفردها الناتج من البرنامج (Output) فإن

هذه اللغة تعرف بأنها لغة دالية (Functional) مثل لغة (HOPE) .

ب - وإذا كانت المدلولات لا تحدد الناتج من البرنامج تماما كما في الحالة السابقة فإن ذلك

يجري علي لغة ليسب (LISP) .

ج - إذا احتوت المدلولات على علاقات (Relationships) مع بعضها البعض فإن ذلك ينطبق تماما علي لغة العلاقات مثل لغة البرمجة المنطقية برولوج (PROLOG)، كما هو مبين بالشكل (١٠-٣).

وبذلك يتضح الفرق في المعالجة بين لغات الذكاء الاصطناعي مثل البرمجة المنطقية والبرمجة بأسلوب القائمة (برولوج و ليسب) واللغات التقليدية مثل فورتران وباسكال مع ملاحظة مايلي:

أ - يمكن استخدام لغات الأمر (فورتران وباسكال) لحل مشكلة معينة في الذكاء الاصطناعي بدون وصف أو وضع اي شئ عن الحل المطلوب ولكن يتم وضع الخطوات اللازمة لايجاده .

ب - وبالعكس فانه يجري استخدام لغات الذكاء الاصطناعي التطبيقية ذات العلاقات والدالية في إيجاد الحل المطلوب، وذلك من خلال عملية تحديد شكل الحل، وبعد ذلك يتولي البرنامج معالجة البيانات أو المعارف لتحديد الحل نفسه، وبذلك يكون الحل موصف أو معرف من خلال البرنامج .

ج - يلاحظ عدم جمود التعريف والاستخدام لهذه اللغات وقصرها فقط علي عمل معين، فيمكن استخدام لغة البرمجة بأسلوب القائمة لعمل برامج انضباطية (itrative) بينما يمكن استخدام لغة باسكال او لغة بيسك في عمل برامج للذكاء الاصطناعي، لأنها سوف تكون غير مناسبة تماما.

د - لتوضيح الفرق بين هذه اللغات فإننا نورد المثال التالي :
لحساب مضروب الرقم (n!) والمعرف بالمعادلة الآتية:

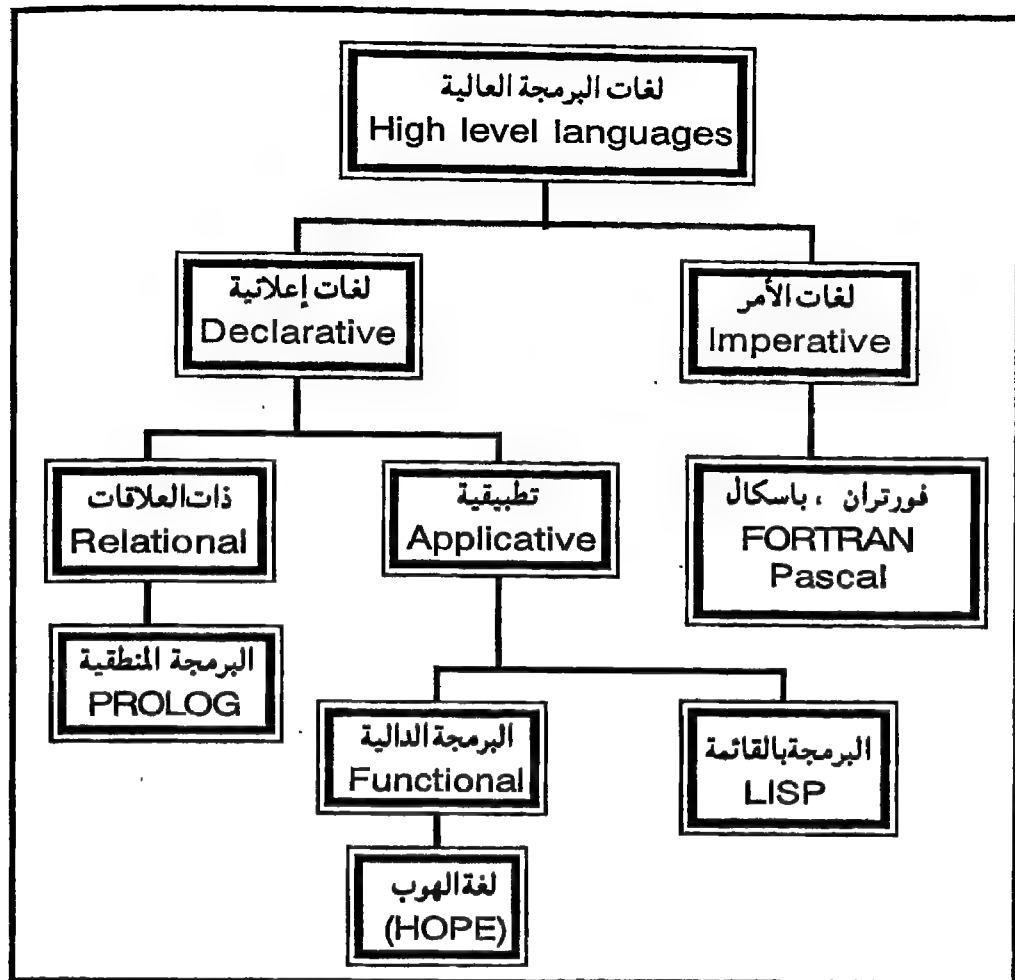
$$n! = n(n-1)(n-2)(n-3) \dots [n-(n-2)]1$$

أولا : - باستخدام لغة باسكال مع استخدام التكرار (recursion) كمايلي:

```
FUNCTION factorial ( n : integer) : integer;
BEGIN
    IF n=1 THEN factorian : = n
    ELSE factorial : = n * factorial(n-1);
END; { factorial }
```

ثانيا : - باستخدام لغة البرمجة بأسلوب القائمة (LISP) واستخدام التكرار .

```
(defun factorial (arg)
  (cond
    (= arg 1 ) 1 )
  ( t ( * arg ( factorial ( - arg 1 ) ) ) ) )
```



شكل (١٠-٣) تصنيف لغات البرمجة

يلاحظ التشابه الواضح بين اللغتين في حساب المضروب، إلا أنه يفضل لغة البرمجة بإسلوب القائمة (LISP) لتطبيقات الذكاء الاصطناعي نظراً لإحتوائها ملامح أكثر مناسبة، مثل طرق ربط المتغيرات علاوة على القدرة على استخدام القوائم .

ثالثاً : - باستخدام لغة البرمجة المنطقية برولوج كمايلي:

```

factorial ( 1 , 1 ) .
factorial (Input , Result ) : -
  lmin1 is Input-1
  factorial ( l min1 , Facmin1);
  Result is Input * Facmin1).
    
```

مع ملاحظة أنه يمكن إضافة علاقات أخرى تمنع التكرار إلى ما لانهاية. ومن الواضح أن

طبيعة المشكلة المراد حلها هي التي تحدد اللغة المناسبة للإستخدام. وبذلك تعتبر لغتي برولوج وليسب اللغتين الأساسيتين لبناء نظم الذكاء الإصطناعي، فبينما تحتوى لغة برولوج على تقنيات التوحيد (Unification) للوصول الى الحل فإن لغة ليسب ولغة (OPS5) تحتويان علي تقنيات السلاسل المتفرعة الى الأمام والى الخلف (Forward / backward chaining)، ولقد ظهرت لغات كثيرة معتمدة على التوليف بين اللغات المختلفة، مثال ذلك لغة (POP11) والتي نشأت من توليف لغتي ليسب وباسكال، وتعتبر التطوير الحديث للغة ليسب، وتستخدم علي الحاسبات الكبيرة مثل (VAX)، وتحتوى على محرر تحاوري (Interactive Editor)، ومنفذ تزايدى (Incremental Compiler)، وتتشابه التراكيب لهذه اللغة مع لغة باسكال مع الاستعانة بقواعد العمل بالقوائم من ليسب حيث يمكن ان يطلق عليها لغة توافق بصمات القوائم (List Pattern Matcher) .

الفصل الحادى عشر

تطبيقات باستخدام
اللغات المنطقية
(البرولوج السريع)

**Applications Using
Logic Languages
(Turbo Prolog)**

يتناول هذا الفصل شرح برامج تطبيقية للذكاء الاصطناعي وكيفية تمثيل المعارف والأوجه المختلفة للتقنيات التي أوردت في الأبواب السابقة.

(١١-١) لغة البرمجة المنطقية (Prolog)

ظهر وتطور النمط الأول من هذه اللغة في جامعة مارسيليا حيث قام بالتطوير (Alain Colmeraure) في السبعينات من هذا القرن، وتعتبر هذه اللغة إحدى أدوات التنفيذ في برامج الذكاء الاصطناعي ونظم الخبرة وخصوصا التطبيقات الهندسية، حيث تم بناء كثير من الغلافات (Shells) ونظم الخبرة مثل (Xi, APES, ESP/Advisor) والتي أظهرت تفوق هذه اللغة على لغات كثيرة مثل لغة باسكال. ولقد كان لسهولة بناء برامج بهذه اللغة تحمل صفة الذكاء وتتألف مع المستخدم (User Friendly) وراء تقدم هذه اللغة. وفي عام ١٩٨١ طورت اليابان خطة قومية طموحة لإنتاج الجيل الخامس من الحاسبات وتم اختيار لغة البرمجة المنطقية برولوج اللغة الاساسية لهذا الجيل من الحاسبات.

(١١-٢) لغة البرمجة المنطقية السريعة (Turbo Prolog)

تعتبر لغة البرمجة المنطقية السريعة إحدى لغات الجيل الخامس وتعتمد أساسا على الملامح الأساسية للغة البرمجة المنطقية الا انها تتميز بالسرعة وإستخدام أساليب المنطق الطبيعي للوصول الى حل المشكلات، والعمل على الحاسبات الشخصية كما يمكن إستخدامها في بناء تطبيقات متنوعة مثل:

- ١- النظم الخبيرة (Expert Systems).
- ٢- قواعد المعارف (Knowledge Bases).
- ٣- مدخل المواءمة لمعالجة اللغات الطبيعية (Natural Language Interfaces).
- ٤- نظم المعلومات الادارية الذكية (Smart Information Management Systems).
- ٥- قواعد البيانات الديناميكية المرتبطة بالعلاقات (Dynamic Relational Databaes).
- ٦- بناء الغلافات (Shells) للتطبيقات المختلفة.
- ٧- ايجاد الحلول للمعادلات التفاضلية على أسس رمزية .
- ٨- إثبات النظريات المختلفة.

تختلف لغة البرمجة المنطقية السريعة عن اللغات الاجرائية (Procedural)

الآخري فيمايلي:

- ١ - تعتبر هذه اللغة لغة توصيف للأشياء (Descriptive) يتمثل في ثلاث محاور:
 - أ - المحور الاول : توصيف الاسماء والتركيبات للأشياء او الاهداف المكونة للمشكلة .
 - ب- المحور الثاني : توصيف اسماء وكنه العلاقات الموجودة بين هذه الاشياء او الاهداف .
 - ج- المحور الثالث : توصيف الحقائق والقواعد التي توصف او تحكم هذه العلاقات .

وبذلك تتميز هذه اللغة علي لغات كثيرة مثل لغة باسكال .
 - ٢ - تستخدم اللغة وسائل الاشتقاق والاستدلال لايجاد الحل ، والذي يتم اوتوماتيكيا عند تنفيذ البرنامج الذي يحاول ايجاد جميع الحلول الممكنة ، وبذلك يكون تفاعوريا (Interactive) مع المستخدم .
 - ٣ - تستخدم هذه اللغة ميكانيكية التتبع الخلفي (Backtracking) وذلك بعد الوصول الى أول نتيجة صائبة فانه يقوم بالتتبع الخلفي لايجاد نتائج أخرى او التاكيد من النتيجة الاولى .
 - ٤ - تستخدم اللغة نحوا بسيطا للجمل يختلف عن معظم نحو اللغات الآخري التقليدية .
 - ٥ - تعتبر هذه اللغة من اللغات العليا (High Level) و اكثر علوا من لغة باسكال حيث يمكن القول بان كل سطر من البرنامج يقابل عشرة اسطر تقريبا من برنامج الباسكال لتنفيذ مهمة واحدة .
 - ٦- تحتوي اللغة على امكانية داخلية للتعرف على البصمات (Built-in Pattern Recognition Facility) علاوة على تركيب انضباطى على الكفاءة .
- ان تعلم البرمجة المنطقية السريعة من خلال شرح برامج تعليمية وتطبيقات ذكية سوف يؤدي الى تفهم اساس اللغة واسلوب الذكاء الاصطناعى .

(١١-٣) التمثيل الرمزي للكيانات والعلاقات

(١١-٣-١) التمثيل الرمزي للنشاط الرياضى

- لتصميم وكتابة برنامج بلغة البرولوج السريع يبين العلاقة بين الاشخاص وميولهم للنشاطات الرياضية ، فان البرنامج يتكون من الشرائح التالية :
- ١ - المجالات (domains) يحتوى المجال على توصيف للأشياء او للاهداف المستخدمة فى

البرنامج حيث يجرى تمثيلها علي انها رموز (symbols) وعلى ذلك فإن الشخص ونوع النشاط الرياضي تمثل رمزيا وتكتب كمايلي:

domains

person, activity = symbol

٢- التوصيف لنوع العلاقات الإسنادية (predicates) يحتوى هذا الجزء على توصيف لنوع العلاقة الإسنادية بين رمز الشخص ورمز نوع النشاط الرياضي وفى هذه الحالة تكون العلاقة هى التفضيل وتكتب فى شكل فعل يفضل كمايلي:

predicates

prefers(person, activity)

٣- التعبيرات أو البنود (clauses) تحتوى على:

١- الحقائق (Facts) مثل يفضل (اسم الشخص) نوع النشاط الرياضي الذى يحبه وتكتب كمايلي:

clauses

prefers(mohamed, tennis).

prefers(ahmed, football).

prefers(adel, swimming).

prefers(khaled, tennis).

٢- القاعدة (Rule) مثل يفضل مصطفى (X) اذا فضل عادل (X) حيث (X) هى متغير يدل على نوع النشاط وتكتب هذه القاعدة فى الشكل:

prefers(mostafa,X) if prefers(adel,X).

وبذلك يصبح البرنامج المتكامل كمايلي:

domains

person, activity = symbol

predicates

prefers(person, activity)

clauses

prefers(mohamed, tennis).

prefers(ahmed, football).

prefers(adel, swimming).

prefers(khaled, tennis).

prefers(mostafa,X) if prefers(adel, X).

٤- إحراز الهدف (Goal) بعد تحميل البرنامج من خلال نافذة الحوار فان البرنامج سوف يسأل عن نوعيه الحل الصحيح المطلوب فى شكل ما هو الهدف المطلوب كمايلي:

Goal: _

وحين ذلك يتم إدخال الجملة او السؤال الذى يراد به الاستدلال ومعرفة حقيقة ما مثل

هل يفضل مصطفى السباحة مع عدم وجودها صراحة فى التعبيرات المدخلة:

```
prefers( mostafa , swimming )
```

فإن البرنامج سوف يجيب من خلال نافذة الحوار أن هذا صحيح (True). ويمكن شرح ذلك بأن البرنامج قد أستخدم قواعد التوحيد المذكورة في الفصل الثامن من هذا الكتاب وذلك برط المتغير (X) فى القاعدة:

```
preferes(mostafa,X) if preferes(adela,X).
```

مع الحقيقة:

```
preferes(adela, swimming).
```

ليقرر يستدل على أن مصطفى يفضل السباحة، ويطلب البرنامج إحراز هدف آخر وذلك بكتابة:

Goal: _

فإذا كتبنا ما يلى:

```
preferes (mostafa , tennis).
```

فإن البرنامج يجيب بالنفي وأن هذا غير صحيح (False)، ويطلب البرنامج إحراز هدف آخر

Goal: _

فإذا كتبنا (Individual) كمتغير فى الشكل:

```
preferes( Individual , tennis )
```

فإن البرنامج سوف يستدل على أسماء الاشخاص الذين يفضلون لعب التنس ويجيب:

Individual = mohamed

Individual = khaled

2 Soutions

Goal : _

(Variables) المتغيرات (١١-٣-٢)

تبدأ الاسماء المتغيرة بحروف كبيرة (Capital) متبوعة بأحرف كبيرة او صغيرة (Small) او يمكن تمثيلة بالخط الذي يكتب اسفل الحرف (" - ") وبذلك تمثل (X) متغيرا كما يلى فى القاعدة السابقة :

```
preferes(mostafa,X) if preferes(adela, X).
```

وعلى ذلك فإن

```
My_first_correct_variable_name
```

```
Sales_LQ_LL_8b
```

تمثل متغيرات، اما اذا بدأت بحروف صغيرة فإنها لا تمثل متغيرات مثال :

```
second_attempt
```

```
disaster
```

(Objects and Relations) الأشياء والعلاقات (١١-٣-٣)

واضح من البرنامج السابق ان :

١ - أسماء الاهداف تبدأ بحروف صغيره (Small) متبوعة بأى عدد من الحروف او الارقام او الخط السفلى (Underscore).

٢ - أسماء العلاقات تبدأ بحروف صغيرة و يمكن أن تضم أحرف او اعداد او خطوط سفلية مثل قملك سوزان حصاناً ويأكل خالد اللحم وغير ذلك.

owns(susan, horse).
eats(khaled, meat).
valuable(gold).
car(mercedes, blue, station_wagon).

ومن الواضح فى المثال السابق ان المجالات والإسنادات لا يجب ان تحتوي على أرقام كمايلي:
domains
person, activity = symbol
predicates
prefers (person, activity)

(١١-٤) الاهداف المركبة (Compound Goals)

(١١-٤-١) قاعدة بيانات للسيارات المستعملة

لتصميم برنامج يحتوى علي قاعدة بيانات للسيارات المستعملة تشتمل على نوع السيارة وعدد الكيلومترات التى قطعتها السيارة والعمر واللون والتمن مثلاً، فإن البرنامج يمكن أن يتكون من الاجزاء الرئيسة الآتية:

١- المجالات (domains): تحتوى فى هذه الحالة علي ثلاثة أسطر : يشتمل السطر الاول على تعريف مجال الماركة او النوع (Brand) واللون(colour) على انهما رموز(symbols) ويحتوي السطر الثانى على توصيف مجال عمر السيارة (age) والتمن(price) على أنهما أرقام صحيحة (integers)، ويعرف السطر الثالث نطاق عدد الكيلومترات المقطوعة (kilometerage) على أنها أرقام حقيقية (real) ربما تكون خارج نطاق الأعداد الصحيحة، ويمكن ان تحتوى على العلامة العشرية.

٢- التوصيف للعلاقات الاسنادية (predicates): يجرى استخدام كلمة سيارة (car) على انها العلاقة التى تربط بين النوع وعدد الكيلومترات المقطوعة والعمر واللون والتمن، والتى سوف تتغير من سيارة الى اخرى وتكتب بالترتيب الذى لا بد ان يتم المحافظة عليه بعد ذلك كمايلي:

car(brand,kilometerage, age, colour, price)

٣- التعبيرات او البنود (clauses): تشتمل على البيانات التفصيلية، وتكتب بالترتيب

الوارد في العلاقات الاسنادية كما هو واضح في الجملة الاولى والتي تشتمل على سيارة دوجان شاهين قطعت ١٣٠٠ ميل وعمرها ٣ سنوات ولونها احمر وثمانها ١٢٠٠٠ جنية. وبذلك يصبح البرنامج كما يلي:

domains

brand, colour = symbol

age, price = integer

kilometerage = real

predicates

car(brand, kilometerage, age, colour, price)

clauses

car(dogan shahin, 13000,3, red, 12000).

car(ford, 90000,4, gray,25000).

car(datsun, 8000,1,red,30000).

٤- إحراز هدف (Goal) : بعد تحميل البرنامج وطلب البرنامج احراز هدف وذلك من خلال

Goal : _

نافذة الحوار كما يلي:

فإذا اردنا السؤال عن السيارة التي ثمنها ٢٥٠٠٠ جنية فإننا نكتب :

car(Make,Kilometerage,Years_on_road, Body, 25000).

ويجب البرنامج :

Make=dogan sahin, Kilometerage=90000,

Years_on_road=4, Body=gray

1 soulution

Goal : _

وإذا حاولنا إيجاد الاجابة على السؤال التالي : هل هناك فى قاعدة البيانات سيارات ثمنها

اقل من ٢٥٠٠٠ وتكتب فى شكل طلب تحقيق للأهداف المركبة Compuond Goals

من شقين كما يلي:

car(Make Kilometerage,Years_on_road,Body,Cost)and
Cost < 25000.

فى هذه الحالة فإن البرنامج سوف يعطى بيانات جميع السيارات التى يقل ثمنها عن

٢٥٠٠٠ .والتي تبين ان اللغة مزودة بعلاقة اقل من (<) وتكون الاجابة :

Mark=dogan sahin, Kilometerage=13000

Years_on_road=3,Color = red .

(١١-٤-٢) المتغير المجهول (Anonymous Variable)

يعرف المتغير المجهول بانه المتغير الذى لايراد الاستدلال عليه من البرنامج، وفى هذا المثال

وعندمحاولة الاستعلام عن جزء من البيانات مثل الثمن والعمر فقط فإنه يكتب بكتابة شرطة

سفلية فى مكان المتغير المجهول والذي لا نريد الاستعلام عنه ويكتب كما يلي:

car(,Age,_,Cost) and Cost < 27000.

فإن البرنامج سوف يعطى كل السيارات التى يقل ثمنها عن ٢٧٠٠٠ ويجب :
 Age = 3, Cost = 12000
 Age = 4, Cost = 25000
 2 Solutions
 Goal: _

ويمكن استخدام المتغيرات المجهولة للتعبير عن الحقائق مثل:

owns(,shirt).
 washes(,).

والتي تعبر عن المعنى يملك كل واحد قميص او كل واحد يغتسل وهى حقائق.

(١١-٥) التتبع الخلفى (Backtracking)

(١١-٥-١) التمثيل الرمزي للصفات الشخصية

لشرح ميكانيكية التتبع الخلفى فإننا نورد قاعدة بيانات تبين العلاقة بين أسماء الاطفال وأعمارهم فى فصل دراسى، ويمكن للبرنامج ان يأخذ الشكل الآتى :

domains
 child = symbol
 age = integer
 predicates
 pupil(child, age)
 clauses
 pupil(mohamed ,9).
 pupil(ahmed,10).
 pupil(khaled,9).
 pupil(mostafa,9).

يتم التعامل مع أسماء الاطفال على أنها رموز، حيث يشتمل السطر الثانى فى البرنامج على تعريف مجال العمر على انه عدد والذي يتكون من المجالات التى تحتوى على سطرين يشتمل الاول على تعريف مجال صحيح (integer) ويحتوي تعريف العلاقة الاسنادية التلميذ (pupil) والتي تربط بين اسم الطفل وعمره pupil (child, age) ثم يلى ذلك البنود على شكل قاعدة بيانات بأسماء الاطفال واعمارهم .

pupil(mohamed ,9).
 pupil(ahmed,10).
 pupil(khaled,9).
 pupil(mostafa,9)

لشرح ميكانيكية التتبع الخلفى فإننا نطلب من البرنامج استخراج اسماء كل طفلين يبلغ عمر

كل منهما ٩ سنوات وذلك بكتابة :

pupil(Person 1,9) and
pupil(Person 2,9) and
Person 1 <> Person 2.

والذى يفيد ان البرنامج سوف يقوم بالاستدلال عن كل طفلين مختلفين عمر كل منهم ٩ سنوات ويجيب البرنامج بمايلى:

person 1 = mohamed, person 2 = khaled
person 1 = mohamed, person 2 = mostafa
person 1 = khaled , person 2 = mohamed
person 1 = khaled , person 2 = mostafa
person 1 = mostafa , person 2 = mohamed
person 1 = mostafa , person 2 = khaled
6 Solutions

Goal :

يلاحظ ان البرنامج قد قام فى أول الأمر بمقارنة أول طفل عمره ٩ سنوات (محمد) بجميع الاطفال بقاعدة البيانات وتم الوصول الى ان خالد ومصطفى لهم نفس العمر، ثم بدأ فى المرحلة الثانية بمقارنة الطفل الثانى بالقائمة والذى يبلغ عمره ٩ سنوات (خالد) بالاطفال من أول القائمة مرة ثانية الى ان وجد ان محمد ومصطفى يبلغ كل منهما ٩ سنوات، ثم رجع البرنامج ثانيا لمقارنة الطفل الاخير فى البرنامج (مصطفى) بجميع الاطفال من أول القائمة مرة ثالثة الى ان وصل الى محمد وخالد.
ومعنى ذلك ان البرنامج يمكنه التتبع من الخلف الى أول البرنامج لعقد المقارنة مع شرط فى اخر البرنامج.

(١١-٦) النفى باستخدام الاداة (لا) (Tautology Using Not)

(١١-٦-١) التوافق بين العادات للأشخاص (Matchmaker)

من أهم صفات البرمجة المنطقية السريعة هى استخدام اداة النفى لا والتى يتبين دورها فى هذا البرنامج الذى يهدف الى ايجاد التوافق بين العادات المكتسبة (Matchmaker) للأشخاص المختلفين وذلك لايجاد أنسب الاختيارات لعقد الزواج بين الرجل والمرأة.
يتركب البرنامج من المجالات (domains) والتى يجرى فيها تعريف الشخص على انه رمز ثم ياتى تعريف الإسنادات (predicates) فى شكل حقائق تؤكد عادات وصفات الاشخاص. مثال ذلك ان يكون الشخص مدخنا او نباتيا وأن فاطمة يمكن ان تتزوج احد الاشخاص كمايلى:

```

person = symbol
predicates
    male(person)
    smoker(person)
    vegetarian(person)
    fattmah_could_marry(person)
goal
    fattmah_could_marry(X) and
    write(" a possible choice for fattmah is " , X ) and nl.
    Clauses
    male(mohamed).
    male(ahmed).
    male(khaled).
    smoker(aly).
    smoker(khaled).
    vegetarian(mohamed).
    vegetarian(khaled).
    fattmah_could_marry(X) if male(X) and not(smoker(X)).
    fattmah_could_marry(X) if male(X) and vegetarian(X).

```

ومن الملاحظ أن قاعدة البيانات التي تمثل التعبيرات أو البنود (Clauses) تحتوي علي العادات المختلفة واسماء الاشخاص ويمكن وضع القاعدة أن فاطمة يمكن ان تتزوج شخص ما بصفات تحدد عند احراز الهدف، وبذلك تأتي شروط احراز الهدف كمايلي:

```

fattmah_could_marry(X) and
write(" a possible choice for fattmah is " , X ) and nl.

```

والتي تبين ان فاطمة يمكنها الزواج من الرجل (X) وبدل السطر الثاني علي ان البرنامج يمكن أن يكتب اسم الرجل في سطر جديد. ولشرح استخدام لا المنطقية فاننا نطلب الشرط ان فاطمة يمكن ان تتزوج شخص لا بدخن وان يكون نباتيا كما هو موضح فيمايلي:

```

fattmah_could_marry(X) if male(X) and not(smoker(X))
fattmah_could_marry(X) if male(X) and vegetarian(X).

```

وعند تحميل البرنامج فانه سوف يجيب بأن فاطمة يمكنها ان تتزوج محمدا.

```

a possible choice for fattmah is mohamed.

```

(١١-٦-٢) تمثيل وبرمجة العلاقات العائلية

لتمثيل وبرمجة العلاقات العائلية فان البرنامج يتكون من:

- ١ - المجالات (domains) المحتوية على تمثيل الشخص في شكل رمزي.
- ٢ - توصيف العلاقات الاسنادية (predicates) التي تبين صفة الذكر والانثى للشخص وان العلاقات مثل الاب والام والاخ والاخت والعم والجد تقع بين شخصين .
- ٣ - التعبيرات أو البنود (clauses) التي تحتوي علي الحقائق المشتملة على الاسماء وكذلك

الشروط التى تربط بين جميع افراد العائلة كما هو مبين فى التعليقات المصاحبة للبرنامج
(comments) والتى تشرح كل شرط على حدة كمايلي:

domains

person = symbol

predicates

male(person)
female(person)
father(person, person)
mother(person, person)
parent(person, peson)
sister(person, person)
brother(person, person)
uncle(person, person)
grandfather(person, person)

clauses

male(mohamed).
male(mostafa).
male(aly).
male(ahmed).
female(fattmah) .
female(soad).
female(marwa).
female(rania).
mother(marwa, fattmah).
mother(mohamed, rania).
father(mohamed, aly)
father(fattmah, mostafa).
father(soad, aly).
father(marwa, mohamed).
parent(X,Y) if mother(X,Y).
parent(X,Y) if father(X,Y).
brother(X,Y) if /*The brother of X is Y if */
male(Y) and /* Y is a male and */
parent(X,P) and /*the parent of X is P and */
parent(Y,P) and /*the parent of Y is P and */
X <> Y. /* X and Y are not the same */
sister(X,Y) if /*The sister of X is Y if */
female(Y) and /* Y is female and */
parent(X,P) and /* the parent of X is P and */
parent(Y,P) and /*the parent of Y is P and */
X <> Y. /* X and Y are not the same */
uncle(X,U) if /*the uncle of X is U if */
mother(X,P)and /* the mother of X is P and */
brother(P,U). /*the brother of P is U. */
uncle(X,U) if /*The uncle of X is U if */
father(X,P) and /*the father of X is P and */
brother(P,U). /*the brother of P is U */
grandfather(X,G) if /*The grandfather of X is G */

```

father(P,G) and      /*if the father of P is G */
mother(X,P).         /*and the mother of X is P.* /
grandfather(X,G) if /*The grandfather of X is G */
father(X,P) and      /*if the father of X is P   */
father(P,G).         /* and the father of P is G */

```

وعند تحميل وتنفيذ البرنامج من خلال نافذة الحوار فإن البرنامج يمكن ان يجيب على معظم الأسئلة التي تدخل في تكوين هذه العائلة مثل هل محمد اخ لأحمد ؟

brother(mohamed,ahmed)

ومن يكون جد مروة ؟

grandfather(X , marwa)

ويمكن وضع اسئلة تكون اكثر تعقيداً من الاسئلة السابقة وهكذا . ومن خلال العرض السابق، يتضح ان البرامج الذكية فى لغة البرمجة المنطقية السريعة تتكون من الآتى:

١- المجالات (domains) العلاقات الإسنادية (predicates) والتعبيرات أو البنود (clauses) وطلب تحقيق الاهداف (goal) وتأخذ الشكل :

```

domains
/* ... domain statement ... */
predicates
/* ... predicate statements ... */
goal
/* ... subgoal_1 , subgoal_2, etc. */
clauses
/* ... clauses ( rules and facts ) ... */

```

٢- الحقائق (Facts) والشروط أو القواعد (Rules) وتأخذ الشكل :

```

relation( object , object , ... , object )
relation( object , object , ..., object ) if
relation( object , ..., object ) and
relation( object , ... , object ) .

```

ولكى تنطبق لغة البرمجة المنطقية السريعة (TP) مع الانماط الاخرى من لغة البرمجة المنطقية (prolog) فإنه يمكن كتابة (:-) بدلا من اذا (IF) وكذلك (،) بدلا من (و) المنطقية (and) كما هو واضح فى المثال التالى:

```

is_older(person1, person2) :-
age( person1 , Age1 ) ,
age( person2, Age2 ) ,
Age1 > Age2 .

```

٣ - المتغيرات وتنقسم الى نوعين:

أ - المتغير الحر (Free variable) وهو المتغير الغير محدد القيمة.

ب - المتغير المحدد (Bound Variable) وهو المتغير المحدد بقيمة ثابتة.

(١١-٦-٣) تمثيل وبرمجة الهوايات الشخصية: يبين البرنامج الاتى العلاقة بين الشخص ونوع الهواية التى يفضلها كما يلى:

domains

person , hobby = symbol

predicates

likes(person,hobby)

clauses

likes(mohamed,reading).

likes(aly,computers).

likes(aly,jogging).

likes(ahmed,jogging).

likes(samy,swimming).

likes(samy,reading).

عند محاولة سؤال البرنامج عن احرار الهدف المركب (compound goal) فى الشكل :

likes(X, reading) and likes(X , swimming)

فإن المتغير (X) يعبر عن قيمة غير معرفة ويبدأ البرنامج فى البحث فى الحقائق من اليسار الى اليمين ومن اعلى الى اسفل لكى يحصل على قيمة (X) فى الجزء الاول من الهدف المركب (likes(X, reading) الى ان يصل الى النتيجة ان (X) هى محمد ويجب :

likes(mohamed , reading) .

فى نفس الوقت فإن البرنامج يضع مؤشرا (pointer) فى قاعدة البيانات تبين المدى الذى وصل له فى عمليه البحث حيث ان (X) اصبحت الان معرفة بـ محمد ويبدأ البحث بعد ذلك عن الجزء الثانى من الهدف المركب :

likes(X , swimming)

وذلك من بداية المؤشر الى آخر قاعدة البيانات الى ان يصل الى النتيجة :

likes(samy,swimming)

(١١-٧) المجالات القياسية

يبين الجدول انواع المجالات القياسية المستخدمة فى لغة البرمجة المنطقية السريعة، وعلى ذلك فإن الثلاثة أسطر الآتية متكافئة فى انشاء سطر جديد

write(' \ 13')

write(' \ n')

nl

وعند استخدام المجالات القياسية فإنه يمكن الاستغناء عن كتابتها فى المجالات وكتابة تعريف الإسنادات كمايلي:

predicate

alphabet _ position(char,integer)

والذى يعطى الحروف الابدجية ورتبتها مثل:

alphabet _ position(' a ' , 1)

alphabet _ position(' b ' , 2)

بهذه الطريقة يمكن تمثيل عملية جمع اعداد صحيحة أو اعداد حقيقية مثل :

add(X , Y , Z)

والتي تحمل معنى:

$X+Y = Z$

او كتابتها كمايلي :

add(integer,integer,integer)

add(real,real,real)

من هنا نري ان اللغة تسمح بتعدد التعريف للعلاقات الإسنادية :

multiple predicate_declaration

(١١-٧-١) استخدام المجالات القياسية

يتناول هذا البرنامج تصميم دليل مصغر للتليفونات : Mini Telephone Directory

والذى يستخدم التعريف القياسى للإسنادات مثل (readln , write) والذى يحتوى على

قاعدة بيانات للاسم ورقم التليفون ويعطى البرنامج رقم التليفون عند إدخال الاسم. مثال :

reference(" mohamed " , Y)

ويتكون البرنامج من الآتى:

predicates

reference(symbol,symbol)

goal

write("please type A name: "),

readln(the_name),

reference(the_Name , phone_No),

write("the phone number is " , phone_No),nl.

clauses

reference("mohamed", "2451892").

reference("ahmed", "5697672").

reference("soad ", "2674006").

reference("marwa ", "03-991051).

The phone number is 245189

ويجب البرنامج :

char	'a', 'b', 'z'
integer	From -32.768 to +32.767
real	42705 , - 9999 86.72 , -911,929 , - 52 e238 , - 79.83 e+21
string	" mohamed's book "
symbol	telephone , " railway_ticket "
\Number	a chracter with ASCII value Number
\n	New line character
\t	Tabulate character .

يبين البرنامج التالي كيفية استخدام المجال الحرفي (char domain) كمايلي:

predicates
isletter(char)

clauses
isletter(ch)if ch <= 'z' and 'a' <= ch .
isletter(ch)if ch <= 'z' and ' A' <= ch.

isletter('%').
isletter('Q').

والذي يتم فيه إحراز أهداف كمايلي :

(٨-١١) تمثيل توصيف الاشياء المركبة (Compound Objects)

عند إستخدام أشياء أو كيانات مركبة فانه يجرى تمثيلها ومعاملتها معاملة الشيء نفسه،
مثال ذلك : يملك محمد حصان يطلق عليه صفة الاسود :

owns(mohamed,hourse(black))

أو يقرأ احمد كتاب الايام للدكتور طة حسين :

reads(ahmed , book("AL_AIYAM",Taha Hussin)

فإن الشيء المركب في المثالين السابقين هما :

horse(black)

book("AL_AIYAM", " Taha Hussin ")

ويطلق علي الجزء الأول من الشيء المركب (الحصان او الكتاب) على انه الدليل

(Functor) ثم يليه بقية الاجزاء الاخرى المكونة للشيء نفسه وتكتب بالشكل الاتي:

functor(object1 , object2 , ... , object N)

ويمكن اتباع مايلي فى تعريف الاشياء المركبة (compound objects) فمثلاً إذا فرضنا ان هناك علاقة التسلك مثل : owns(mohamed , X) فإن المتغير (X) يمكن ان يمثل طبقا

للتعريف السابق على أنه رمز وبذلك يكون التعريف :-

owns(name , article) كالاتى لإسناد التملك

وبذلك تم ادخال كلمة (article) التى يمكن وصفها بالمجالات :

domains

article = book(title,author) ;

horse(colour)

title , author , colour = symbol

ولقد تم استخدام (أ و) المنطقية (OR) فى شكل (;) وبذلك تم تمثيل وصف

الكتاب بالعنوان والمؤلف ووصف الحصان بصفة اللون، ويمكن وضع تصور آخر لتوصيف إسنادات

التملك (owns predicate) لتمثيل الحقائق التالية :

owns(mohamed , book ,("AL_AIYAM" , "Taha Hussin")) .

owns(mohamed , horse(blacky)) .

owns(mohamed , bankbook(1000)) .

owns(mohamed , thing) . فإذا طلب إحراز هدف فى الشكل :

والذى يقابل المجالات :

article = book(title,author) ; horse(colour) ;

bankbook(integer) .

فإن البرنامج يجب :

Thing = book("AL_AIYAM " , " Taha Hussin ")

Thing = horse(blacky)

Thing = bank book(1000)

قاعدة : لتعريف المجال للأشياء أو الكيانات المركبة يمكن كتابة :

domains = alternative1(D , D , ...) ;

alternative2(D , D , ...) ;

حيث تمثل البدائل alternative1 , alternative2 دلائل الشئ functors ويمثل

(D,D,...) أسماء أو صفات أو مجالات قياسية مثل (integer, real,char) مع ملاحظة مايلي:

١ - وضع علامة (أو) المنطقية (;) بين البدائل .

٢ - ان يتكون البديل من دليل (functor) وقائمه من المجالات الموصفة للشئ .

يبين البرنامج الاتى دليل (functor) لتحريك نقطة الادخال (cursor) على الشاشة تنفيذا

لتوصيف العلاقة الإسنادية cursor(row , column) كمايلي :

domains

row, column , step = integer

```

movement = up(step) ; down(step);
left(step); right(step)
predicates
  move_cursor(row,column,movement)
clauses
  move_cursor(R,C,up(step)) :-
    R1=R-step,cursor(R1,C).
  move_cursor(R,C,down(step)) :-
    R1=R+step, cursor(R1,C).
  move_cursor(R,C,left(_)) :-
    C=C-1 ,cursor(R,C1)) :-
  move_cursor(R,C,right(_)) :-
    C1=C+1,cursor(R,C1).

```

يتكون البرنامج من المجالات التى تحتوى على تعريف السطر والعمود وخطوة التحرك (step) على انها اعداد صحيحة، وكذلك تعريف إتجاه التحرك على انه الى اعلى او اسفل او يمين او يسار وذلك من خلال خطوة التحرك وتعريف إسنادات التحرك على انها صف وعمود وحركه، ثم التعبيرات او البنود التى تحتوى على تفاصيل التحرك، ويمكن تحريك نقطة الادخال مثلا من الصف الرابع والعمود التاسع على الشاشة بمقدار عمودين الى اعلى وذلك بطلب احراز الهدف:

move_cursor(4 , 9 , up(2)) .

(١١-٨-١) المجالات المشتعلة علي تفصيلات دقيقة للكيانات المركبة

كما ذكرنا سابقا من استخدام المجال :

```
book(title , author)
```

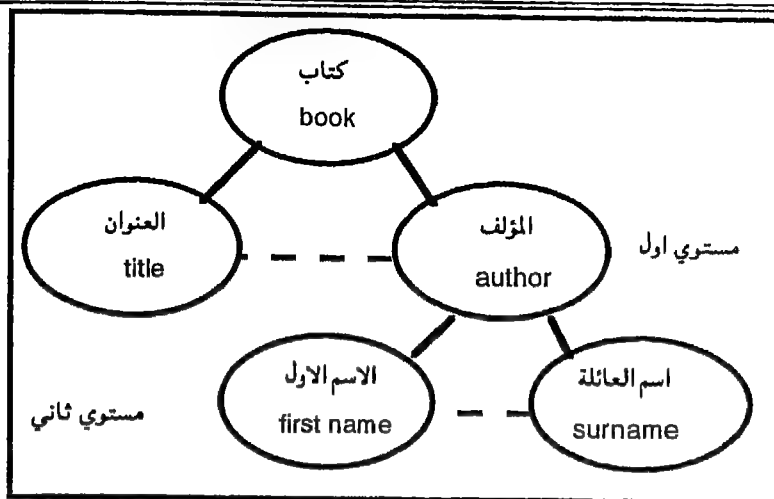
والتي توصف المؤلف على انه رمز (symbol) لاتفى بمتطلبات تعريف التفاصيل الدقيقة عن المؤلف مثل الاسم الاول (firstname) واسم العائلة (surname) وبذلك يصبح المؤلف من الاشياء أو الكيانات المحتوية على تفاصيل اخرى .بذلك يمكن كتابة المجالات التى تشمل تفاصيل وصف المؤلف بالشكل التالى :

```
domains
  articles = book( title , author ) ;
  author = author( firstname , surname )
  title , firstname , surname = symbol
```

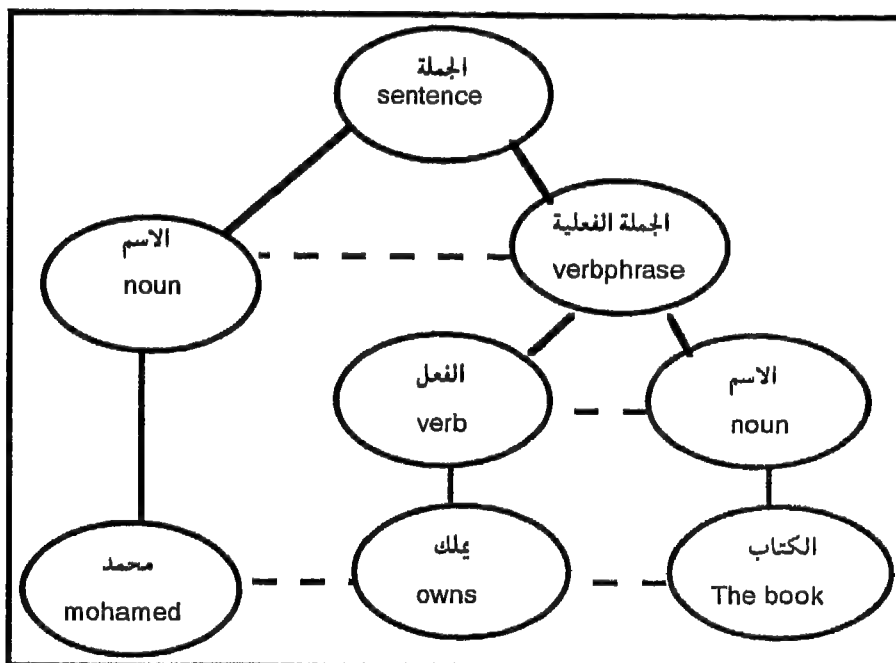
يمكن استخدام التمثيل الشجرى (Tree Representation) لوصف الاشياء التى تحتوى على اشياء اخرى ذات تفاصيل دقيقة كما هو مبين فى شكل (١١-١) وفى هذه الحالة فإن كل سطر من المجالات يقابل مستوى واحد من الشجرة :

مثال آخر : اذا أردنا تمثيل الجملة التالية :يملك محمد الكتاب.

فان التمثيل الشجرى لهذه الجملة يصبح كما فى شكل (١١-٢) . بذلك يمكن كتابة :



شكل (١١-١) التمثيل الشجري للكيانات المركبة



شكل (١١-٢) التمثيل الشجري للجملة الفعلية

sentence = sentence(noun , verbphrase)

noun = noun(word)

وبذلك تكون المجالات :

verbphrase = verbphrase(verb , noun) ; verb(word)

verb = verb(word)

ويكون توصيف العلاقات الاسنادية للمجالات كمايلي :
 sentence(noun(mohamed),verbphrase(verb(owns),noun(book)))

(٩-١١) التكرار (Recursion)

عند المعالجة الرمزية يستخدم التكرار فى حالتين :

١ - عند توصيف العلاقات (relations) بمساعدة العلاقات نفسها والذي يسمح بتكرار العلاقة اكثر من مرة .

٢ - عند توصيف شئ مركب (compound object) يكون جزءاً من شئ مركب آخر ويلزم التكرار .

يبين البرنامج التالى حساب مضروب الرقم N! من خلال المعالجة الرمزية وذلك من خلال طلب احراز الهدف الذى يمثل حقيقة وقاعدة فى نفس الوقت :

factorial (N , F)

والذى سوف يكون صحيحا (true) عندما تكون (F) مساوية لقيمة N! ، أى عندما

$$F = N * (N - 1) * (N - 2) * \dots * 3 * 2 * 1 .$$

تكون :

ويصبح البرنامج :

domains

n=real

f=real

/* REAL number used for answer

so answers greater than 32767

can be returned */

predicates

factorial(n,f)

clauses

factorial(1,1).

factorial(N,Res) if

N > 1 and

N1 = N - 1 and

factorial(N1,ResN1) and

Res = N * ResN1.

فإذا طلبنا احراز الهدف :

factorial(2 ,Answer)

فإن البرنامج سوف يستخدم القاعدة

factorial(2,Res) if 2>1,N1=2-1,factorial(N1,FacN1),

Res=2 * Fac N1 .

وعلى ذلك فإتة لابد من حساب :

factorial (1 , Fac N 1) .

وذلك باستخدام الحقيقة :

factorial(1,1)

ثم حساب النتيجة Result :

Res = 2 * Fac N 1

وبذلك تكون النتيجة تساوى (2)، وعند طلب احراز هدف :

goal : factorial(4 , Answer)

فإن البرنامج سوف يستدعى الآتى :

calls : factorial(4 , Res)

calls : factorial(4 - 1 , Fac N 1) , 4 * Fac N 1

calls : factorial(4 - 1 - 1) , FacN 11 , (4 -1) * FacN11 .

calls : factorial(4-1-1-1, Fac N111),(4 -1-1) * FacN111

calls : factorial(1 , 1)

أى ان البرنامج يبدأ أولا بحساب مضروب (١١) ثم يستدعى العلاقة مرة أخرى لحساب

(٢١) ثم مرة ثالثة لحساب (٣١) ومرة أخرى لحساب (٤١) وهكذا.

(١١-٩-١) الاشياء المتكررة (Recursive Objects)

تستخدم وسائل التكرار في هذه اللغة لتوصيف الاشياء التى تحتوى على عدد من العناصر

الغير معلومة مسبقا، ومثال ذلك عند تصميم برنامج يصف أسماء التلاميذ فى فصل من الفصول

بدون المعرفة المسبقة لعدد التلاميذ فى الفصل وذلك من خلال ايجاد قائمة بالاسماء والذى يمكن

تحقيقه بالآتى:

١- يجرى توصيف المجال بقائمة فارغة تمثل الفصل وهو فارغ كمايلى :

Classlist = empty

٢- تستخدم طرق التعريف المتكرر (recursive defintion) الذى يدل على احتواء القائمة

للأسماء :

classlist = class(name,classlist)

فإذا كان بالفصل تلميذ واحد (محمد) فإن :

class(mohamed , X)

الاسم محمد سوف يكون اول إسم بالقائمة، وتمثل (X) القائمة الفارغة؛ فإذا كان هناك طفل

اخر (احمد) فإن القائمة سوف تكون :

```
class( mohamed , class( ahmed , empty ) )
```

وهكذا فإن التعريف النهائي للمجال المركب والمتكرر كمايلي:

```
classlist = class( name , classlist ) ; empty
```

(١١-١٠) القوائم (Lists)

تعتبر القوائم من أساسيات تراكيب البيانات في لغة البرمجة المنطقية السريعة حيث يتم توصيفها بسهولة كمايلي:

١ - تكتب عناصر القوائم داخل قوسين مربعين مع استخدام الواو المنطقية في الداخل كالآتي:
[.., ..] مثل [1,2,3] التي تمثل قائمة للارقام و (cataloge) والتي تمثل قائمة من الحيوانات.

٢ - عند توصيف قوائم تحتوي على أعداد صحيحة فإنه يمكن وصف المجالات كمايلي:

domains

```
integerlist = integer *
```

وتدل علامة (*) على ان هذه القائمة تحتوي على عناصر كثيرة تبدأ من الصفر.

٣ - يمكن للعنصر داخل القائمة ان يكون اى شئ او قوائم اخرى بشرط ان تتبع نفس المجال، وتعرف هذه العناصر بعد ذلك كما في الجدول شكل (١١-٣).

domains

```
objectlist = objects*
```

```
objects =.....
```

٤- تنقسم القوائم الى جزئين الرأس (head) والذيل (tail) ويعرف الرأس بأنه أول عنصر في القائمة ويعرف الذيل بأنه الباقي من القائمة، كما هو موضح بالجدول شكل (١١-٣).

٥- تستخدم اللغة العمود (|) لفصل الرأس عن الذيل للقائمة، وبذلك تكتب القائمة المكونة من X,Y كالآتي: [X | Y] وتستخدم ايضا في التوافق (Matching) بين القوائم، كما في الجدول المبين في شكل (١١-٤).

(١١-١٠-١١) استخدام القوائم (Using Lists)

١- البحث عن عنصر في قائمة (عضوية القائمة) (List Membership)

إذا فرضنا القائمة التالية والمكونة من أسماء :

```
[ mohamed , ahmed , mostafa , khaled]
```

وأردنا أن نبحث عن إسم من هذه الاسماء بالقائمة فانه يجرى تعريف العلاقة الاسنادية عضو

List	Head	Tail
[(a),(b),(c)]	' a '	['b ' , 'c ']
[1]	1	[] an empty list
[]	undefined	undefined
[[1,2,3], [2,3,4]]	[1 , 2 ,3]	[2 , 3 , 4]

شكل (٣-١١)

List 1 القائمة ١	List 2 القائمة ٢	variable Binding تحديد قيمة المتغير
[X, Y, Z]	mohamed , eats , ice-cream	X= mohamed , Y = eats , Z = icecream .
[7] [1 , 2 , 3 , 4]	[X Y] [X, Y Z]	X = 7 , Y = [] X = 1 , Y = 2 Z = [3 , 4]
[1 , 2]	[3 X]	لا يمكن ان تتم المقارنة لإختلاف الراس في القائمتين

شكل (٤-١١)

كما يلي : `member(name,namelist) .`

ويبين البرنامج الآتي طريقة البحث عن اسم داخل القائمة، حيث يبدأ البحث في رأس القائمة أو لامع اهمال الذيل وذلك من خلال السطر الاول في التعبيرات (البود) والذي ينص على

الآتي: `member(Name, [Name | _])` .

فاذا كان الاسم المطلوب هو رأس القائمة، فان البحث ينتهى عند ذلك، وإذا لم يكن الاسم المطلوب في رأس القائمة فان الجملة الثانية :

`member(Name , [_ | Tail]) if member(Name ,Tail)`

تقود البرنامج الى البحث عن الاسم في الذيل، وبذلك يتم البحث في القائمة ككل، مثال ذلك

مايلي: domains

`namelist = name *`
`name = symbol`

predicates

`member(name, namelist)`

clauses

`member(name,[name|_]).`

`member(Name,[_|Tail]) if member(Name, Tail).`

٢- استخدام وكتابة عناصر القائمة : لكتابة عناصر القائمة فى أسطر منفصلة ومتتالية، فاننا نستخدم العلاقة التكرارية الآتية:

```
write_a_list[ ]
write_a_list( [ Head | Tail ] ) if
write( Head ) ,nl ,write_a_list(Tail) .
```

حيث تعمل الجملة الاولى على ان ينتهى البرنامج (stop) عن العمل اذا اصبحت القائمة فارغة [] ، وتعمل الجملة الثانية على ان يكتب الراس فى سطر والذيل فى السطر الذى يليه.

٣- ضم القوائم : تستخدم العلاقة الإسنادية (append) لضم قائمتين فى قائمة واحدة مثال ذلك :

```
append(List1,List2,List3)
```

والتي تعنى ضم القائمة الاولى (List1) مع القائمة الثانية (List2) لتكون القائمة الثالثة (List3). مثال ذلك :

```
append( [ 1 , 2 , 3 ] , [ 5 , 7 , 9 ], L )
```

فتكون النتيجة القائمة

```
L= [1,2,3,5,7,9]
```

واذا كانت القائمة الاولى (List1) فارغة ولاحتوى على عناصر [] ، فان القائمة الثالثة تأخذ قيم القائمة الثانية أى:

```
append([ ],List2,List2)
```

و اذا تم اختيار المتغير (X) كرأس لكل من القائمة الاولى والقائمة الثالثة:

```
List 1 = [ X | L 1 ]
```

```
List 3 = [ X | L 3 ]
```

ويستخدم البند

```
append( [ X | L 1 ] , List 2 , [ X | L 3 ] ) if
append(L 1,List 2,L3)
```

بذلك يتكون الذيل (L3) من ذيل القائمة الاولى (L1) والقائمة الثانية كلها والمتبقى من القائمة الثالثة. ويشرح البرنامج التالى ميكانيكية استخدام العلاقة الإسنادية (append) كما فى المثال الآتى:

```
domains
integerlist = integer*
predicates
append(integerlist,integerlist,integerlist)
writelist(integerlist)
clauses
```

```

append([ ],List,List).
append([X|L1], List2, [X|[3]]) if
append(L1,List2,L3).
writelist([ ]).
writelist([ ]).
writelist([ Head | Tail ])
write(Head) ,n1, writelist(Tail) .

```

(Search Mechanism) ميكانيكية البحث (١١-١١)

تنقسم ميكانيكية البحث في هذه اللغة الى جزئين:

- ١ - الجزء الأول : ويشتمل على عملية التوافق بين الاهداف المطلوب احرازها وبين التعبيرات أو البنود المحتوية على الحقائق والقواعد :

(matching a goal with a clauses)

وتعرف هذه العملية بعملية التوحيد (Unification process)

- ٢ - الجزء الثاني : ويشتمل على دراسة كيف تتحكم اللغة في البحث عن الحلول (search for solution) لاحراز او تحقيق الأهداف.

(١١-١١-١) عملية التوحيد (Unification Process)

كما رأينا سابقاً أن عملية التوحيد من العمليات الأساسية في الذكاء الاصطناعي ويمكن شرحها بالمثل التالي:

(١١-١١-٢) قاعدة بيانات للمطبوعات بالمكتبات

يمكن شرح عملية التوحيد من خلال تصميم قاعدة بيانات للمطبوعات كما يلي:

```

domains
    title,author = symbol
    pages        = integer
    publication = book (title , pages )
predicates
    written_by(author , publication )
    long_novel ( publication )
clauses
    written_by(el-sharkawy , book( "EL-ARRD " ,, 210 ).
    written_by(el-sadat, book ( "EL-BAHS AN EL-THAAT",600)).
    long_novel(book(Title,Length)):-
        written_by(_ ,book(Title,Length)) ,length>300 .

```

يشتمل البرنامج علي قاعدة بيانات للمطبوعات بالمكتبات والتي تتركب من المجالات التي تشتمل على تعريف العنوان (title) الأرض أو البحث عن الذات والمؤلف (Author) الشراوى

أو السادات، على أنهم رموز، وتعريف عدد الصفحات على أنها أعداد صحيحة (integers) وتعريف المطبوعة (Publication) على أنها تمثل كتاب مكون من عنوان وعدد صفحات. وتعرف العلاقة الإسنادية بين المطبوعة والمؤلف على أنه العلاقة كتب بواسطة (written by)، وأن المطبوعة هي قصة طويلة (long novel)، وتشتمل البنود على البيانات لكتابي الأرض والبحث عن الذات كحقائق:

```
written_by(el-sharkawy , book( "EL-ARRD ",210 ).
written_by(el-sadat,book( "EL-BAHS AN EL-THAAT", 600)).
```

وكذلك القاعدة :

```
long_novel(book(Title,Length)):-
written_by(_ , book (Title,Length)),
length > 300 .
```

فإذا طلب من البرنامج إحراز الأهداف التالية :

١ - إذا اتخذ الهدف الشكل الآتي:

```
written_by(X,Y)
```

حيث (Y) , (X) تمثلان متغيرات حرة (Free variables)

وتجرى عملية التوحيد بمقارنة (Y) , (X) في الهدف المطلوب مع الجملة المشتملة على الحقائق واحدة تلو الأخرى حيث يتم ربط قيم المتغيرات كمايلي:

```
X =el-sharkawy , Y= book( "EL-ARRD " , 210 )
```

للبنود الأول وكذلك :

```
X = el-sadat , Y = book( "EL-BAHS AN EL-THAAT " ,600 )
```

في الحالة الثانية، وبذلك تم ربط قيم المتغيرات بالقيم داخل الاقواس وذلك لتطابق الدليل (written by) في الهدف والبنود.

٢ - إذا اتخذ الهدف الثاني الشكل الآتي:

```
written_by( X , book( "EL-BAHS AN EL-THAAT " , Y ) ) .
```

فإن البرنامج يبدأ في المقارنة بين الهدف المطلوب و أول بند كمايلي:

```
written_by( X , book( "EL-BAHS AN EL-THAAT " , Y ) )
written_by(el-sharkawy , book( "EL-ARRD " , 210 ) )
```

ولاتتم عملية التوحيد نظرا لاختلاف عنوان الكتاب في الحالتين. ويتولى البرنامج عقد التوافق مع الجلسة الثانية:

```
written_by(el-sadat, book("EL-BAHS AN EL-THAAT",600)
```

وفي هذه الحالة فإن قيم المتغيرات تصبح :

```
X = el-sadat
Y = 600
```

حيث يتطابق عنوان الكتاب في الحالتين. يتضح ان عملية التوحيد بين هدف وجملة من البنود تكون ناجحة اذا كان:

- ١- الدليل (Functor) واحد في الحالتين.
- ٢ - عدد الحدود متساوى داخل الأقواس المكونة للشئ المركب ويجرى التوحيد بين كل زوجين متقابلين على حدة.

ويمكن تلخيص عملية التوحيد كما يلي:

- ١ - يتم توحيد المتغير الحر مع اي شق من التعبير أو البند.
 - ٢ - يتم توحيد العدد الصحيح بنفس قيمته أو بمتغير حر.
 - ٣ - يتم التوحيد بين عناصر جملتين مركبتين اذا كان المدلول واحداً في الحالتين وعدد الحدود داخل الاقواس متساوي، وتتم عملية التوحيد بشكل زوجي.
- وبذلك فإن عملية التوحيد هي التحديد لقيم المتغيرات وتراكيب البيانات من خلال توافق عام للبصمات (Pattern - Matching Mechanism).

(١١-١٢) التحكم في البحث لاحراز الاهداف

(Controlling the Search for Soultions)

لشرح كيف تتحكم تقنيات البرمجة المنطقية السريعة في البحث للوصول الى الحل فإننا نورد البرنامج الآتي والذي يربط بين الاشخاص وهواياتهم والمبين فيما يلي:

```
domains
    name, thing = symbol
predicates
    likes(name,thing)
    reads(name)
    is_inquisitive(name)
clauses
    likes(mohamed,juice).
    likes(khaled,tea).
    likes(Z,books) if
        reads(z) and
        is_inquisitive(z).
    likes(khaled,books).
    likes(khaled,films).
    reads(mohamed).
    is_inquisitive(mohamed).
```

يتكون البرنامج من المجالات التي تبين أن الاسم والشئ يمكن أن يمثل رمزيا ، كما تحتوي الإسنادات على علاقة التفضيل بين الإسم والشئ، أو أنه سوف يتم قراءة الاسم reads

(name) وان هذا الاسم سوف يتم الاستعلام عنه :

is_inquistive (name)

تتكون البنود من الحقائق التي تبين حب محمد للعصير وحب خالد للشاي، ثم القاعدة ان المتغير المجهول (Z) الذي يحب الكتب، مع الاستعلام عن اسمه من البنود التالية لهذا الشرط.
likes(Z , books) if
reads(Z) and
is_inquistive(Z) .

ولنا ان نحاول احراز الهدف

likes(X , Juice) and likes(X , books)

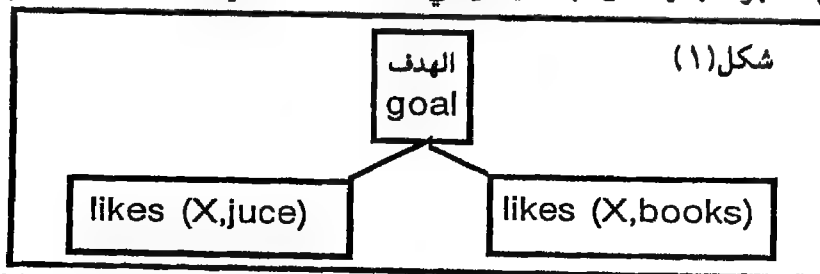
والذي يتركب من جزئين. ولنبين كيف يعمل البحث فإنه يتم استخدام التمثيل الشجري للأهداف شكل (١) كمايلي: عند بداية البحث فان البرنامج سوف يبحث في التمثيل الشجري للهدف مستخدما القاعدة الاساسية التي تنص على: "ان إحراز أجزاء الهدف يتم من اليسار الى اليمين وان يراعى الترتيب الوارد في الاسنادات والبنود بالبرنامج، على أن يتم وضع خط منقوط تحت أجزاء الهدف الذي يتم احرازه وكتابة الجملة المطابقة تحت هذا الخط ". وبذلك فان البرنامج سوف يقوم بإحراز الجزء الأول من الهدف كما هو مبين بالتمثيل الشجري شكل (٢) حيث يتطابق الجزء الأول من البنود للهدف مع الحقيقة الأولى، وتحديد قيمة المتغير (X) بالاسم محمد. عند البحث لتنفيذ الجزء الثاني من الهدف، فانه لا يتم التطابق نظرا لان قيمه (X) محددة بمحمد واختلاف كلمة العصير عن الكتب، وبذلك فان البحث يستمر ويحاول مطابقة البند الثاني ولكن لا يتم التطابق لتغير الاسم الى خالد وبذلك يذهب البرنامج الى البند الثالث :

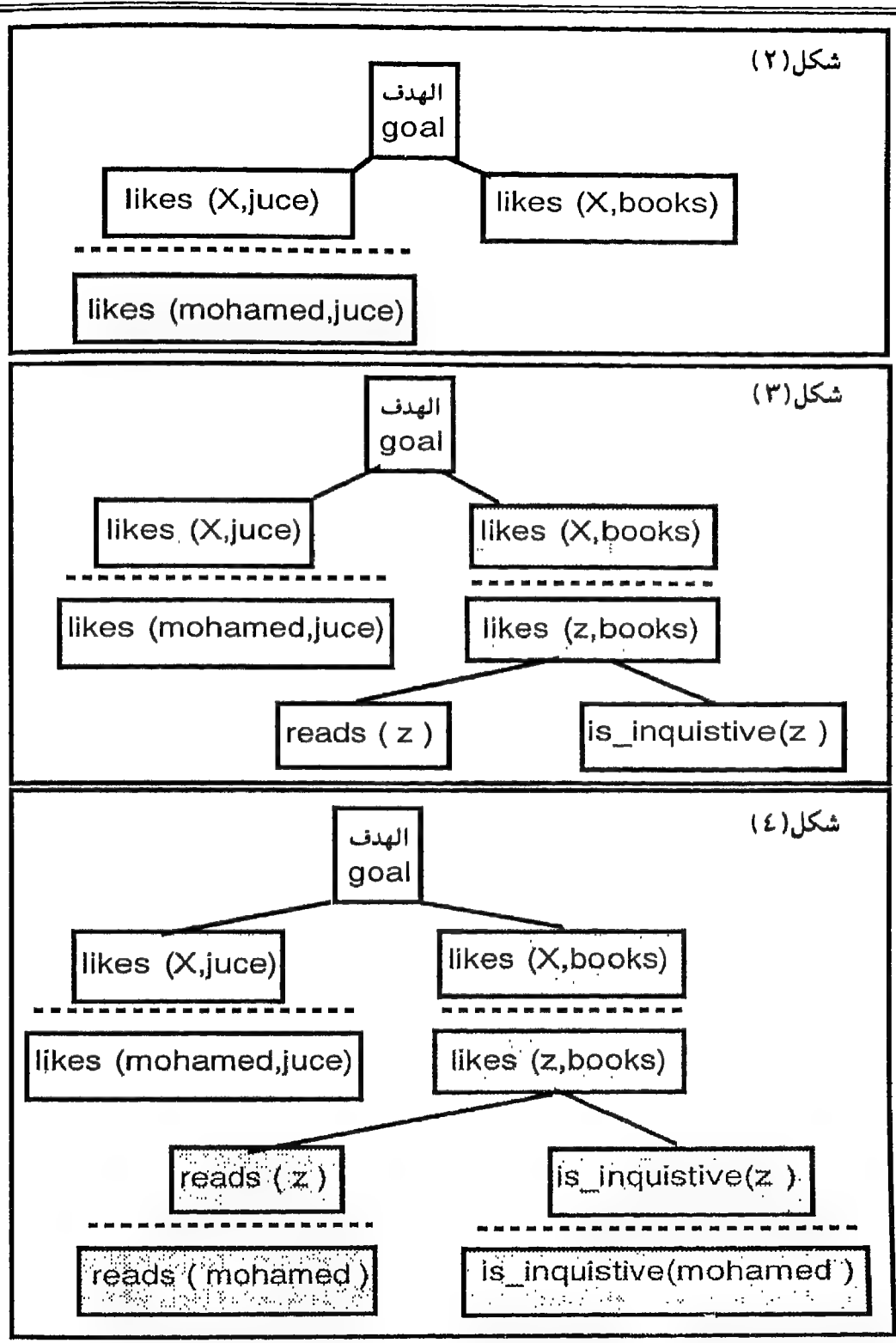
likes(Z , book) if reads(Z) and is_inquistive(Z).

حيث يتم تطابق الجزء الأيسر من الجملة اولا وبذلك تتطابق قيم (Z) , (X) أى يتم التحول من المتغير (X) وابدالة بالمتغير (Z) كما هو مبين في شكل (٣) :وبذلك أصبح التمثيل الشجري للهدف محتويا علي جزء جديد هو :

reads(Z) and is_inquistive(Z)

وتنفيذا لهذا الجزء الجديد فان البحث يصل الي ان محمدا هو الذي يحب الكتب ويصبح التمثيل





الشجري شكل (٤)؛ وبذلك يمكن القول بأنه قد تم إحراز هدف عندما إمتد البحث ليشمل أطراف التفرع للتمثيل الشجري (الاوراق). وبذلك فان البحث يسير في اتجاهات مختلفة معتمداً علي البداية طبقاً للهدف المطلوب احرازه. كما رأينا ان البرنامج يظل يبحث لاحراز الجزء الثاني من الهدف بعد ان يتحقق الجزء الأول منه وانه سوف يبدأ في التتبع الخلفي (Backtracking) لايجاد حل اخر بديل للحل الذي وصل اليه او عندما لا يتم احراز اي جزء من الهدف.

إحراز او تحقيق هدف (Subgoal) :

لإحراز جزء من هدف او هدف فان البرنامج يبدأ بالبحث عن تطابق الجزء الاول من الهدف (من ناحية اليسار) مع الجملة الأولى لقاعدة البيانات (clauses) وذلك في اتجاهين :
الاتجاه الاول : اذا تم العثور على رأس اول جملة (Head) مطابقة فان البرنامج سوف يقوم بالخطوات التالية :

- ١ - تحديد مكان هذا الرأس بمؤشر (Pointer) وذلك لتحديد النقطة التي يمكن أن يبدأ منها التتبع الخلفي للبحث عن جملة اخرى تطابق هذا الجزء من الهدف.
 - ٢ - تحديد قيمة المتغير الحر بالرموز الواردة في هذا الرأس. بذلك تصبح المتغيرات محددة.
 - ٣ - اذا كانت الجملة المطابقة تمثل الجزء الايسر (left side) من القاعدة ، فان هذه القاعدة لا بد لها أن تنطبق، وذلك بإفتراض أن الجزء الأيمن منها (right side) يمثل هدفاً جديداً.
- الاتجاه الثاني : اذا لم يتم العثور علي جملة مطابقة ولم يتم إحراز هدف (goal fails) فان البرنامج سوف يحاول ان يتتبع خلفياً (backtracks) لايجاد الحل البديل للهدف السابق. وبذلك فإن جميع المتغيرات المحددة من الهدف السابق تصبح حرة مرة ثانية، ويبدأ البرنامج البحث من بداية النقطة التي تم تحديدها بمؤشر (pointer) في الحالة الاولى، ويبدأ البرنامج الرجوع الي الخلف عدة مرات محاولاً احراز أجزاء الأهداف الاخرى حتى ينتهي، واذا لم يحصل على نتيجة، فان البرنامج يؤكد انه لم يتم احراز أية أهداف .

(١١-١٣) التتبع الخلفي المدفوع (Forced Backtracking)

يمكن استخدام العلاقة الاسنادية عدم النجاح في احراز الهدف (fail predicate) وذلك لإجبار ودفع البرنامج على التتبع الخلفي، وبين البرنامج الآتي هذا الاستخدام .

```
domains
    name = symbol
predicates
    father(name, name)
    everybody
```

clauses

father(name,name)
everybody

clauses

father(mohamed,marwa).
father(ahmed,aly).
father(ahmed,soad).
everbody if
father(X,Y) and
write(X,"is " , Y,"s father \n") and
fail.

يتكون البرنامج من المجالات التي تشتمل علي التمثيل الرمزي للاسم والعلاقات الاسنادية التي توضح ان العلاقة التي تقوم بين اسمين وان كل شخص يمكن ان يكون ابا، وتبين البنود حقائق الأبوة التي تربط بين إسمين معرفين والقاعدة التي تنص على: " ان كل شخص يمكن ان يكون ابا إذا كانت هناك علاقة أبوة بينه وبين شخص آخر (X,Y) وان النتيجة هي عدم النجاح في احراز الهدف (Fail)." كمايلي:

everybody if father(X , Y) and
write (X , " is " , Y . " 's father\n") and
fail .

father (X , Y)

وعند تحميل البرنامج لاحراز الهدف :

فان البرنامج سوف يجيب:

Goal : everybody
mohamed is marwa's father
ahmed is aly's father
ahmed is soad's father
False

من الواضح ان البرنامج بعد أن حدد السطر الاول بأن محمد يكون أب مروة قد استخدم كلمه عدم احراز النجاح للتبع الخلفي مره اخرى ليأتى بالحقيقه الثانيه أن أحمد والد علي ثم يتبع خلفيا للمرة الثالثه ليأتى بالحقيقه أن أحمد والد سعاد. وبذلك فان كلمه (fail) قد استخدمت لدفع البرنامج الي التبع الخلفي، وذلك تمشيا مع الجزء الأيمن من القاعدة المبينه سابقا.

(١١-١٣-١) الايقاف المحدد للتبع الخلفي (Preventing Backtracking)

يستخدم عنصر القطع (cut) والذي يجري تمثيله بعلامة التعجب (!) لاييقاف التبع الخلفي عند نقطة محددة، وذلك تمشيا مع القاعدة التي تنص على مايلي:

" لايمكن التبع الخلفي مرورا بعلامة القطع "

ويستخدم القطع في الأحوال الآتية :

١ - عندما تشير الاحتمالات أن فرص ايجاد الحل باستخدام التبع الخلفي غير واردة وبذلك

يؤدي إستخدام الايقاف الي توفير وقت تنفيذ البرنامج .
 ٢ - عندما يكون منطق تصميم البرنامج لايتطلب التتبع الخلفي.
 ويمكن شرح استخدام القطع لايقاف التتبع الخلفى فى المثال التالى:
 اذا كانت هناك ثلاث بنود (r1,r2,r3) التى تشرح نفس علاقه الاسنادية (r) وطلب
 احراز الأهداف التاليه:

a, b , c

فاذا تم كتابة الهدف في شكل القاعدة التالية :

r1 if a and b and ! c.

والتي تعنى أنه لاداعى لبدء التتبع الخلفى من أول البرنامج بل يتم التتبع من بداية علامه
 القطع (!) الي آخر البرنامج .ويمكن إبراز عمل علامة القطع (!) فى البرنامج التالى والذى
 يحمل فكرة ان الرجل يمكن ان يختار المرأة التى توافق فى العادات، وذلك للزواج حيث يقوم
 البرنامج بالاستدلال على كل زوجين متوافقين.

domains

name , sex,interest = symbol
 interest = interest*

predicates

findpairs
 person(name, sex,interests)
 member(interest,interests)
 common_interest (interests,interests,interest)

clauses

findpairs if person (Man, m, ILIST1) and
 person(woman , f ,ILIST2) and
 common_interest (ILIST1 , ILIST2 , _) and
 write(Man , " might marry " , woman) and n1 and
 fail .

findpairs:- write (" -----end of the 1st ----").

common_interest(IL1, IL2, X) if
 member(X , IL1) and member(X , IL2) and !
 person (mohamed , m ,[travel, books,cinema]).
 person(soad , f , [Juice , books , swimming]).
 member(X , [X | _]).
 member(X , [_ | L]) if member (X , L) .

يتركب البرنامج من المجالات التى تحتوى على التمثيل الرمزي للاسم والجنس والعادات وأن

العادات المفضلة للشخص يمكن ان تكون معدومة او أكثر من واحدة. Interests = interest *

كما تحتوى العلاقات الاسنادية على علاقة الإيجاد لزوجين (findpairs) وعلاقة الشخص
 (person) التى تربط الاسم والجنس والعادات والعلاقة (member) التى تربط العادات
 والعلاقة (common-interst) التى تربط العلاقات المتماثلة.

تحتوى البنود على القاعدة التى تنص على إيجاد زوجين من القائمة الاولى (List1) للرجال والقائمة الثانية (List 2) للمرأة اذا كان هناك عادة واحدة مشتركة على الأقل مع كتابة اسم الرجل واسم المرأة واستخدام علاقة النفي فى إحراز الهدف (fail) لكى يصبح التتبع الخلفى واجبا كمايلي:

```
findpairs if person (Man, m, ILIST1 ) and
person( woman , f , ILIST2 ) and
common_interest ( ILIST1 , ILIST2 , _ ) and
write( Man , " might marry " , woman ) and n1 and
fail .
```

وتحتوى البنود كذلك علي القاعدة الاتية:

```
common_interest( IL 1 , IL2 , X ) if
member( X , IL 1 ) and member( X , IL2 )
and ! .
```

والتي تتولى البحث عن العادة المشتركة (X) نى القائمتين، و تشتمل كذلك على أمر الايقاف للتبع الخلفى (!) والتي تمنع تكرار الاسم عدة مرات اذا كان هناك اكثر من عادة واحدة مشتركة.

البنود المحددة والايقاف (Clauses Determinism and Cut)

كما رأينا ان الاسنادات (predicates) يمكن ان تنفذ البنود التي تأخذ أحد الشكليات:

١- البنود أو الجمل المحددة Deterministic Clauses والتي تعطى حلا واحد بدون التتبع الخلفى.

٢- البنود أو الجمل الغير محددة (Non-Deterministic Clauses) وهى الجمل القادرة علي ايجاد حلول متعددة باستخدام التتبع الخلفى، ولذلك يجب اتخاذ الحيلة عند استخدامها حفاظا على وقت تنفيذ البرنامج وحجم الذاكرة المستخدمة، ولذلك يجب التأكد من وجود الجمل الغير محددة قبل بداية تنفيذ البرنامج وذلك باستخدام الامر-check (determ) الذى يؤدي الى إكتشاف وتحديد البنود الغير محددة.

يمكن تحويل الجمل الغير محددة الى جمل محددة باستخدام أمر الايقاف عن التتبع الخلفى لتصبح الجملة محددة بأول حل تصل اليه كما فى الجملة المحتوية على القاعدة الآتية:

```
verify _ member ( X , [ X ] ) : - !
verify _ member ( X , [ _ | Y ] ) : -
verify _ member ( X , Y ) .
```

حيث أن وجود علامة الايقاف (!) سوف تؤدي الى ايقاف التتبع الخلفى.

الفصل الثانى عشر

المعالجة الرمزية
للعمليات الحسابية
والرسوم التصويرية

**Symbolic Processing for
Arithmetic Operations
and Graphics**

(١٢-١) المعالجة الرمزية للعمليات الحسابية

(Symbolic Processing for Arithmetic Operations)

إن قدرات لغة البرمجة المنطقية السريعة علي معالجة العمليات الحسابية مشابهة للغات اخري مثل لغة بيسك (BASIC) ولغة باسكال (PASCAL) ولغة سي (C) ، حيث تتم العمليات الحسابية الأربعة (الجمع (Addition) والطرح (Subtraction) والضرب (Multiplication) والقسمة (Division)) كمايلي:

١ - إذا إحتوي التعبير الاصلي تعبيرات جزئية اخري داخل أقواس فانه يجري حساب هذه التعبيرات الجزئية أولا .

٢ - إذا كان التعبير محتويا علي علامه الضرب (*) او القسمة (/) فانه يلزم حسابها بعد الخطوة الاولى وبدايةً من اليسار الى اليمين.

٣- تشمل الخطوة الاخيرة حساب الجمع (+) و الطرح (-) لما بين الأقواس والحدود للتعبير الكلي. يبين الجدول الآتي شكل (١٢-١) طبيعة ناتج العمليات الحسابية من حيث الشكل العام وبين الجدول شكل (١٢-٢) أولويات إجراء العمليات الحسابية : إذا كان عددا صحيحا (Integer) أو عددا حقيقيا (Real) . لاجراء عمليات المقارنة الحسابية فاننا نورد المثال التالي:

$$X + 4 < 9 - Y$$

والذي يمكن كتابته كما يلي:

plus(X , 4 , value 1)

والتي تعني جمع X , 4 لتعطي القيمة الاولى

الاولوية	العملية
1	+
2	-
3	*
4	/
	mod div
	- + (unary)

النتيجة	الجزء الثاني	العلامة	الجزء الاول
integer	integer	+, -, *	integer
real	integer	+, -, *	real
real	real	+, -, *	integer
real	real	+, -, *	real
real	integer or real	/	integer or real

شكل (١٢-٢) أولويات العمليات الحسابية

mins(9 , Y , value 2)

والتي تعنى طرح Y من القيمة 9 لتعطي القيمة الثانية (value 2)

Less _ than(value 1 , value 2)

والتي تعني ان القيمة (value 1) اقل من القيمة (value 2)

يبين الجدول الاتي شكل (٣-١٢) معاملات عمليات المقارنة وكيفية كتابة علاقاتها.

يمكن عقد المقارنة بين الحروف او المقاطع (Strings) كما يلي: 'a' < 'b'

فان ذلك يكون صحيحا (True) حيث تتم المقارنة بين قيم الحرف في جدول آسكى

المشفرة(ASCII Code) وعند مقارنة أسماء مثل محمد وخالد: khaled > mohamed

فان ذلك يكون غير صحيحا (False) حيث أن قيمة الأسكى لحرف (K) اقل من قيمة

الأسكى لحرف (m) فى جدول آسكى،وعند مقارنة مقطع يحتوى على نفس الحروف الاولى مثل :

" antony " > " antonia "

فإن المقارنة تجري بين كل زوجين من الحروف لهما نفس الترتيب، وفي هذا المثال فان المقارنة

تستمر الي ان تصل الي حرف (y) في الكلمة الاولى والحرف (i) في الكلمة الثانية ويتضح من ذلك

ان قيمة (Y) اعلي من قيمة (i) في جدول آسكى، وبذلك تكون العلاقة صحيحة (True).

وبين البرنامج الآتى كيفية عقد المقارنة لاشياء أو كيانات مركبة (Compound Objects).

domains

d= pair(integer, integer) ; single(integer) ; none

predicates

equal(d , d)

clauses

equal(X , X) .

equal(single (4) , pair(3 , 4) .

equal(pair (2 , 1) , pair(2 , 1)) .

equal(none , none) .

ويشرح البرنامج كيفية كتابة المجالات التي تحتوي علي تعريف الزوجين (d) علي انها

العلامة	العلاقة
<	less than
≤	less than or equal to
=	equal
>	greater than
≥	greater than or equal to
<> or ><	different from

شكل (٣-١٢) معاملات المقارنة للعمليات الحسابية

تمثل زوجين من الأرقام الصحيحة وأن العدد المفرد هو عدد صحيح. تحتوي العلاقة الاسنادية على علامة التساوي $equal(d, d)$ ويمكن طلب احراز الاهداف الاتية :

$$equal(pair(2, 1), pair(2, 1)).$$

لكي يجيب البرنامج بان هذا صحيح .

(١٢-٢) إيجاد الحل لمعادلة الدرجة الثانية

لإستخدام لغة برولوج لإيجاد حل معادلة من الدرجة الثانية فى الشكل :

$$A * X * X + B * X + C = 0$$

$$D = B * B - 4 * A * C$$

حيث يعتمد الحل علي قيم الحد :

وتبعاً لذلك توجد الثلاث احتمالات الآتية للحل:

١ - إذا كانت $D = 0$ فإن الحل يكون وحيداً .

٢ - إذا كانت $D > 0$ فإن الحل يعطي قيمتين X_1, X_2 .

٣ - إذا كانت $D < 0$ فإن الحل يكون منعزلاً لان قيمة (X) غير حقيقية .

ويشرح البرنامج الآتى كيفية إيجاد الحل للمعادلة .

```
predicates
    solve( real , real , real )
    reply( real , real , real )
    mysqrt( real , real , real )
    equal( real , real )
clauses
    solve( A,B,C ) :-
        D = B*B - 4*A*C , reply ( A , B , D ) , nl .
    reply( _ , _ , D ) :- D < 0 , write( " No solution " ) , !.
    reply( A,B,D ) :-
        D = 0 , X = - B / ( 2*A ) , write ( "X=" , X ) , !.
    reply(A,B,D):-
        mysqrt( D , D , sqrtD ) ,
        X1 = ( -B + sqrtD ) / (2*A),
        X2 = ( -B - sqrtD ) / (2*A),
        write ( " X1 = " , X1, " and X2= " , X2).
    mysqrt( X , Guess , Root ):-
        NewGuess = Guess- ( Guess*Guess-X)/2/Guess,
        not (equal(NewGuess, Guess) ) , !,
        mysqr ( X ,NewGuess, Root).
    mysqrt ( _ , Guess, Guess).
    equal(X,Y):-
        X/Y > 0.99999 , X/Y < 1.00001.
```

ويبين البرنامج أن الإسنادات تحتوي علي العلاقة الاسنادية إيجاد الحل (solve) لثلاث

قيم حقيقية، وأن تكون علاقة الإجابة (reply) لثلاث قيم حقيقية وان الجذر التربيعي لقيم

حقيقية وأن علاقة التساوي تنشأ بين قيمتين حقيقتين. يشرح البرنامج البنود في شكل القواعد التالية:

1- القاعدة الأولى : solve(A,B,C) :-

$D = B^2 - 4 \cdot A \cdot C$, reply (A , B , D) , nl .

والتي تحسب قيمة (D) وتنتهي بكتابة قيم (A,B,D)

2- القاعدة الثانية : reply (_ , _ , D) :- $D < 0$, write (" No solution ") , !.

والتي تجيب بعدم وجود حل وتنتهي بعلامة إيقاف (!) للتتبع الخلفي اذا كانت قيمة D سالبة.

3- القاعدة الثالثة : reply(A,B,D) :-

$D = 0$, $X = -B / (2 \cdot A)$, write ("X=" , X) , !.

والتي تطلب حساب (X) في حالة ان (D) تساوي صفرا وكتابة قيمة الحل الوحيد (X) وانهاء التتبع الخلفي.

4- القاعدة الرابعة :

reply(A,B,D):-

mysqrt(D , D , sqrtD) ,
 $X1 = (-B + \text{sqrtD}) / (2 \cdot A)$,
 $X2 = (-B - \text{sqrtD}) / (2 \cdot A)$,
 write (" X1 = " , X1 , " and X2 = " , X2).

والتي تطلب حساب الجذر التربيعي وحساب (X1) , (X2) .

5- القاعدة الخامسة :

mysqrt (X , Guess , Root):-

NewGuess = Guess - (Guess*Guess-X)/2/Guess ,
 not (equal (NewGuess, Guess)) , ! ,
 mysqrt (X , NewGuess, Root).
 mysqrt (_ , Guess, Guess).

والتي تحسب قيمة الجذر التربيعي باستخدام طرق الضبط (iteration) لقيم التخمين

(Guess) تطبيقا للمعادلة : NewGuess = Guess - (Guess*Guess-X)/2/Guess ,

عند تنفيذ هذا البرنامج فاننا نطلب احراز الاهداف التالية :

solve (1 , 2 , 1) .

solve (1 , 1 , 4) .

solve (1 , -3 , 2) .

x = - 1

No solutions

X1 = 2 and X2 = 1

وتكون النتائج كمايلي:

(١٢-٣) الدوال الحسابية (Arithmetic Functions)

تتميز لغة البرمجة المنطقية السريعة عن الأنماط الأخرى بإحتوائها على دوال رياضية وعلاقات إسنادية تعمل بالأرقام الحقيقية والصحيحة كما هو مبين في المجموعات التالية:

١- المجموعة الاولى:

bitand (X , Y , Z) , bitor (X , Y , Z) ,
bitnot (X , Z) , bitxor (X , Y , Z) ,

في هذه المجموعة من الدوال فان قيمة (X) , (Y) لا بد ان تكون اعداد صحيحة، وبذلك يكون الرقم (Z) هو الناتج من اجراء العمليات المنطقية.

XOR , NOT , OR , AND

bitleft (X , N , Y) , bitright (X , N , Y)

٢- المجموعة الثانية

تعطى المتبقى من حاصل قسمة (X) على (Y) وتعطي ناتج قسمة (X) على (Y) حيث (X) عدد موجب.

٣- المجموعة الثالثة :

تشمل دوال حساب المثلثات :

cos (X) , sin (X) , tan (X)

حيث (X) تمثل الزاوية مقاسة بالتقدير الدائري.

٤ - المجموعة الرابعة :

تعطي قيمة (arctangent) لقيمة (X) الحقيقية arctan (X)

العلاقة الأسية (e) مرفوعة للأس exp (X) , (X)

للوغاريتم للأساس e ln (X)

للوغاريتم للأساس ١٠ log (X)

الجذر التربيعي لقيمة X sqrt (X)

حيث يجري تمثيل مابين القوسين بثوابت أو متغيرات تم تحديدها مسبقا.

الفصل الثالث عشر

قواعد البيانات
الديناميكية

Dynamic Databases

(١٣--١) قواعد البيانات الديناميكية (Dynamic Databases)

تعتبر لغة البرمجة المنطقية السريعة لغة استفسارية (Powerful query language) حيث يجري تمثيل قواعد البيانات المرتبطة بالعلاقات (Relational databasis) في شكل مجموعة من الحقائق التي يمكن تحويلها الى قواعد بيانات ديناميكية وذلك بالاضافة أو الحذف. وطبقا لما تم ذكره سابقا فإن عملية التوحيد (Unification) تؤدي الى اختبار الحقائق المتطابقة ووضع القيم للمتغيرات المجهولة وإيجاد بدائل الحلول باستخدام خوارزميات التتبع الخلفي.

في هذا الباب يجري تعريف قواعد البيانات الديناميكية وكيف يمكن إضافة معلومات أو حقائق أثناء تنفيذ البرنامج، وكذلك الحذف للمعلومات والحقائق القديمة أو المضافة، وكيف يمكن زيادة سرعة المعالجة في حالة قواعد البيانات الكبيرة الحجم، ويمكن القول بأن الحقائق الموصفة لقاعدة البيانات الديناميكية تختلف في المعاملة عن الحقائق الموجودة في توصيف العلاقات وذلك لوجودها في جزء خاص من البرنامج.

(١٣-١-١) توصيف قاعدة البيانات

تحتل قاعدة البيانات جزء منفصل يبدأ بالكلمة المفتاح (database) ويأتي بعد الجزء الخاص بتوصيف المجالات (domains) ثم يتوالى بعد ذلك باقي البرنامج كما هو موضح في البرنامج التالي والذي يحتوى على قاعدة بيانات للأشخاص من حيث الاسم والسن والعنوان ونوع الجنس.

```
domains
  name,address = string
  age = integer
  sex = male ; female
database
  person(name,address,age,sex)
predicates
  male(name,address,age)
  female(name,age,sex)
  child(name,age,sex) clauses
  male(Name,Address,Age) if
  person(Name,Address,Age,male).
```

من الواضح أن توصيف قاعدة البيانات يتم باستخدام العلاقة الإنسانية شخص :

```
person(name,address,age,sex)
```

التي تسمح بإمكانية الإضافة أو الحذف لحقائق عن الاشخاص وذلك أثناء تنفيذ البرنامج. وتستخدم العلاقة الإنسانية (asserta) لإضافة وإدخال حقيقة جديدة في بداية الحقائق المدخلة من قبل لعلاقة ما ، وتستخدم العلاقة الاسنادية (assertz) لإضافة وإدخال حقيقة جديدة في

نهاية الحقائق المدخلة من قبل لعلاقة ما. وتستخدم العلاقة الإسنادية (retract) لحذف أية حقائق من قاعدة البيانات، وفيما يلي يتم إضافة حقيقة جديدة عن الشخص محمد وحذف حقيقة عن الشخص أحمد كما يلي :

```
assertz(person("mohamed","Cairo" , 35 ) ) .
retract(person("ahmed",_,_)).
```

بذلك يكون التعديل للبيانات عن طريق الحذف اولا ثم الإضافة. ويمكن حفظ قاعدة البيانات الديناميكية باستخدام العلاقة الإسنادية :

```
save("mydata.dba").
```

والتي تساعد على حفظ الحقائق في ملف كنص (Text file) حيث يتم حفظ كل حقيقة في سطر، ثم يتم الإستدعاء لهذا الملف مرة ثانية بأستخدام العلاقة الاسنادية :

```
consult ("mydata.dba")
```

التي تسمح بقراءة قاعدة البيانات من الملف على الشاشة.

(١٣-١-٢) التعامل مع الحقائق (Handling Facts)

يمكن أن يتم التعامل مع الحقائق المخزونة في الملفات باستخدام العلاقة الإسنادية (readterm) التي يمكنها قراءة أى بيانات أو حقائق تم كتابتها بواسطة العلاقة الإسنادية (write) والتي تأخذ الشكل :

```
readterm(<name>,TermParam).
```

حيث يمثل <name> إسم المجال المستخدم ويبين البرنامج الآتى طريقة عمل هذه العلاقة الإسنادية:

```
domains
    name,addr = string
    one_data_record = p(name,addr)
    file = file_of_data_records
predicates
    person(name,addr)
    moredata(file)
clauses
    person(Name,Addr):-
        openread(file_of_data_records,"dd.dat"),
        readdevice(file_of_data_records),
        moredata(file_of_data_records),
        readterm(one_data_record,p(Name,Addr)).
    moredata(_).
    moredata(File):-not(eof(File)),moredata(File).
```

بشرط أن يحتوى الملف (filedd.dat) على حقائق تتبع وصف المجال المعين مثل :

```
p("Mohamed","EL-TAHREER Street")
p("Ahmed","Tanta Street")
```

ويمكن إستخراج المعلومات من الملف بطلب إحراز الاهداف الآتية والإجابة عليها فى الحوار التالى:

```
Goal : person("Mohamed",Address).
```

```
Address="EL-TAHREER street"
```

1 solution

Goal : person("Mohamed","Not an address").

False

Goal : _

يمكن التعامل مع الحقائق التى توصف العلاقات الإسنادية لقاعدة البيانات فى شكل حدود أو أجزاء (terms) وذلك باستخدام التعريف المجالى لقاعدة البيانات (dbasdom) والذى يتولى تعريف كل حد أو جزء على انه أحد البدائل والمكون من دليل (functor) والتوصيف المجالى لهذا الجزء. مثال ذلك :

database

person(name,telno)
city(cno,cname)

والتي تتحول الى الشكل الآتى بواسطة نظام البرمجة المنطقية :

domains

dbasedom = person(name,telno) ; city(cno,cname)

(١٣-٢) قواعد البيانات الديناميكية الممتدة

تعتبر الحقائق المكونة لقاعدة البيانات جزء من برامج لغة البرمجة المنطقية، وبذلك يعتمد حجمها على حجم الذاكرة المستخدمة (RAM size) ويتم الإضافة أو الحذف للحقائق باستخدام العلاقات الإسنادية (assertz,retract). ولزيادة حجم حقائق قواعد البيانات وتسجيلها على ملفات للبيانات (data files) بالاضافة أو الحذف يمكن إستخدام العلاقات الاسنادية (dbassert , dbretract) وبذلك فإن حجم قاعدة البيانات الديناميكية يعتمد على حجم القرص الصلب المستخدم. يمكن شرح عمل هذه العلاقات الاسنادية فى المثال التالى:

عند تصميم قاعدة بيانات ديناميكية تحتوي على الاسم والعنوان والعمر والنوع (ذكر أو انثى) والهوايات الشخصية فانه يجرى تعريف المجالات وقاعدة البيانات كمايلى:

domains

file = datafile ; indexfile
name , address = string
age = integer
sex = m or f
interest = symbol
interests = interest*

database

person(name, address,age,sex,interests)

حيث تم تعريف ملفين أحدهما للبيانات (datafile) والآخر للفهرسة (indexfile) وتعريف تعدد الهوايات الشخصية بقائمة تبدأ من المحتوي الصفري. وتشمل العلاقات الاسنادية على مايلي:

predicates

dbassert(dbasedom)

```

dbretract(dbasedom)
dbread(dbasedom)
dbass(dbasedom,string,string)
dbaaccess(dbasedom,real)
dbret(dbasedom,string,string)
dbret1(dbasedom,real)
dbrd(dbasedom,string,string)

```

١ - العلاقات الإسنادية لتسجيل وحذف قاعدة البيانات علي الملفات.

٢ - العلاقة الإسنادية للقراءة وللدخول داخل قاعدة البيانات للمعالجة بالاضافة أو الحذف.

٣ - العلاقة الإسنادية لإستخدام الأرقام الحقيقية لتعريف مكان أو موضع الحقيقة أو السلسلة داخل قاعدة البيانات .

تشتمل البنود على مايلي:

dbassert (Term):-	dbass (Term,"dba.ind","dba.dat").
dbretract (Term):-	dbret (Term,"dba.ind","dba.dat").
dbread (Term) :-	dbrd (Term,"dba.ind","dba.dat").

١- قواعد الإضافة أو الحذف أو قراءة البيانات من ملفات البيانات والفهرسة , (datafile , indexfile)

٢- قواعد الأضافة أو التعديل لحقيقة أو سلسلة في ملف البيانات و ربط موضع أو مكان الحقيقة برقم حقيقي في ملف الفهرسة كمايلي:

```

dbass(Term,IndexFile,DataFile) :-
    existfile ( DataFile ) , existfile(IndexFile) , !
    openappend(datafile,Datafile) ,
    writedevic(datafile) ,
    filepos( datafile , pos , O ) ,
    write(Term) , n1 ,
    closefile(datafile) ,
    openappend(indexfile , IndexFile ) ,
    writedevic(indexfile) ,
    closefile(Indexfile) . dbass( Term , IndexFile , DataFile ) :-
    openwrite(datafile , Datafile ) ,
    writedevic(datafile) ,
    filepos(datafile , pos , O ) ,
    write(Term) , n1 ,
    closefile( datafile ) ,
    openwrite(indexfile , Indexfile ) ,
    writedevic(indexfile) ,
    writef("%7.0\n",pos),
    closefile(indexfile) .

```

٣- قواعد غلق الملفات بعد القراءة :

```

dbrd(Term , IndexFile , DataFile) :-
    openread(datafile, DataFile) ,
    openread(indexfile , IndexFile) ,

```

```

    dbaaccess(Term , -1 ) .
dbrd( , , ):-
    closefile(datafile), closefile(indexfile ) , fail .
dbaaccess(Term , Datpos):-
    Datpos>=0,
    filepos(datafile,Datpos,0),
    readdevice(datafile ) ,
    readterm(Dbasedom , Term ) .
dbaaccess(Term , _ ) : -
    readdevice(indexfile ) ,
    readreal(Datpos1 ) ,
    dbaaccess(Term,Datpos1 ) .

```

٤ - قواعد الحذف لحقيقة من قاعدة البيانات وذلك باستخدام وكتابة رقم الفهرسة لهذه الحقيقة بالسالب في ملف الفهرسة كمايلي:

```

dbret(Term,Indexfile,Datafile):-
    openread(datafile,DataFile),
    openmodify(indexfile,IndexFile),
    dbret1(Term, -1).
dbret1(Term,Datpos ) : -
    Datpos>=0,
    filepos(datafile,Datpos,0),
    readdevice(datafile ) ,
    readterm(Dbasedom,Term),I,
    filepos(indexfile,-q,1),
    flush(indexfile),
    writedevise(indexfile),
    writef("%7.0\n",-1),
    writedevise(screen).
dbret1(Term,_) :-
    readdevice(indexfile),
    readreal(Datpos1),
    dbret1(Term,Datpos1).
writef( " %7.0\n",pos),

```

(١٣-٣) المميزات العامة للغة البرمجة المنطقية السريعة

تحتوى لغة البرمجة المنطقية السريعة (T P) عل كل السمات العامة للغة البرمجة المنطقية التي طورت عام ١٩٨١ ، وتعتبر هذه اللغة عملية للأسباب التالية:

- ١ - تساعد اللغة علي كتابة برامج معتمدة علي نفسها للحاسبات الشخصية، وذلك إعتمادا على إمكانيات النوافذ (Windows) والالوان (Colors) والرسوم البيانية (Graphics) حيث يمكن المزج بين النص (Text) والرسوم البيانية داخل النوافذ كما يمكن الولوج (Access) بسهولة الى الذاكرة والتحكم فى المدخل والمخرج علاوة على سهولة التشفير (Coding) .
- ٢ - تمنح اللغة المبرمج فرصة التحكم فى شفرة المنبع (Source Code) وذلك اثناء تجميع

البرامج (Compilation) واثناء التصحيح (Debugging) وذلك بإستخدام إمكانية التتبع (Trace facility) المزود بها اللغة.

٣ - ان اتباع طريقة كتابة البرامج تقدم بيئة فريدة تؤدي الي تقليص الحجم المطلوب للبرنامج.
٤ - تعتبر لغة برمجة متكاملة (Integrated) حيث تسمح بتركيب البرنامج في شكل أجزاء أو نماذج (Modules) منفصلة يتم كتابتها بلغة اخري مثل لغة التجميع او لغة سي وغيرهما (Assembly & C).

٥ - إستخدام الإسنادات القياسية (Standard Predicates) للتعامل مع الملفات والتي تسمح بالولوج العشوائي للملفات.

٦- تحتوي اللغة على الامكانيات الحسابية للاعداد الصحيحة والحقيقية وكذلك المعاملات الحسابية والدوال مثل دوال حساب المثلثات ودوال القطعة (bit-wise) التي تستخدم للتحكم فى حركة الروبوتات.

(١٣-٣-١) عناصر لغة البرمجة المنطقية السريعة

١- الأسماء (Names): تستخدم الأسماء في وصف الثوابت الرمزية والمجالات (Domains) والإسنادات (Predicates) والمتغيرات حيث يتكون الاسم من حرف او شرطة سفلية متبوعا بأحرف أو أعداد، ويشترط ان تبدأ بأسماء الثوابت الرمزية بحرف صغير وأسماء المتغيرات بحرف كبير أو خط سفلي. كما تحتوي هذه اللغة على بعض الاسماء المحجوزة (Reserved Words) والأسماء الممنوع استخدامها (Resricted).

٢- شرائح البرنامج (Program Sections): يتكون البرنامج من الشرائح المتتابعة الآتية:
- تعرف كل شريحة بالكلمة المفتاح (Keyword). يعتبر إحتواء البرنامج على شريحتي الإسنادات (Predicates) وشريحة التعبيرات أو البنود (Clauses) شرطا أساسيا لتكوين البرنامج علاوة علي الاستعانة بشريحة المجالات (Domains) عند تصميم برامج كبيرة الحجم، وإذا لم يتم إدراج شريحة إحراز الهدف (Goal) فان البرنامج سوف يعمل طبقا للغة البرمجة المنطقية القياسية (Clocksin & Mellish) وفي حالة تصميم برامج مكونة من جزئيات فانه يجري إستخدام الشرائح الشاملة (Global) وذلك لتوصيف المجالات أو الإسنادات، ويجب استخدام الكلمة المفتاح (keyword) فى أول كل شريحة. وإحراز هدف واحد عند تنفيذ البرنامج مرة واحدة. وان تتولى التعبيرات التي توصف نفس العلاقات الاسنادية واحدة تلو الأخرى. وعند إحتواء البرنامج علي شريحة قاعدة بيانات (data base) فلا بد أن تأتى قبل شرائح الإسنادات والإسنادات الشاملة.

فيما يلي توصيف للشرائح المكونة للبرنامج.

١ - المجالات (Domains)

تستخدم الطرق الكتابية الآتية لتوصيف المجالات:

١ - الشكل الكتابي لتوصيف مجال ما : $name = d$ حيث تمثل (d) رمز أو

سلسلة أو ارقام حقيقية أو ارقام صحيحة.

٢ - الشكل الكتابي لتوصيف قائمة : $mylist = element\ Dom^*$

حيث يجرى تمثيل القائمة من عناصر مجال قياسي أو معرف بواسطة المستخدم.

٣ - الشكل الكتابي لتوصيف شئ مركب:

$my\ Comp\ Dom = f1(d11, ..., d1N); f2(d21, ..., d2M); ...; fM(dN1, ..., dNk)$

حيث يمثل $f1 =$ الدليل وتمثل $d =$ البدائل الموصفة لاجزاء الشئ المركب.

٤ - الشكل الكتابي لتوصيف مجال ملف :

$file = name\ 1 ; name\ 2 ; ... ; name\ N$

حيث يجرى تسمية الملفات باسماء رمزية.

٢ - الإسنادات (Predicates)

يجرى توصيف العلاقات الإسنادية بالجملة الآتية:

$predname(domain1, domain2, ..., domain\ N)$

والتي تأتي بعد الكلمة المفتاح (predicates)، حيث يمثل اسم العلاقة

الإسنادية (prername) علاقة الربط بين المجالات المختلفة والتي يمكن أن يصل

عددها الى (N) من المجالات، وفي بعض التطبيقات يمكن ان تكون العلاقة

الاسنادية اسم واحد فقط مثل: $choss_teams : _$

$same_league(X, Y)$, $never_played(X, Y)$, $write(X, Y)$.

أو تكون متعددة مثل :

$member(name, namelist)$

$member(number, numberlist)$

حيث تربط العلاقة الاسنادية (member) بين الاسم والقائمة وكذلك بين الرقم الأرقام.

٣ - التعبيرات (البنود) (Clauses)

حيث يمثل التعبير أو البند الحقيقة (fact) أو القاعدة (Rule) التي تشرح إحدى الجمل

الاسنادية والتي يمكن ان تكتب في جملة واحدة أو جملة واحدة متبوعة بأداة الشرط

إذا ($IF = : _$) والأداة المنطقية ($,$) ($AND = ,$) والأداة المنطقية أو ($OR = ;$)

الفصل الرابع عشر

بناء نظم تطبيقية

Building Application Systems

يتناول هذا الباب إستخدام لغة البرمجة المنطقية السريعة (T P) لبناء نظم تطبيقية. ومن خلال إستعراض هذه البرامج يتم للقارىء تفهم ميكانيكية العمل داخل هذه النظم والتي يمكن ان تتخذ نواة فى بناء نظم مماثلة أو أكبر حجما ، تشمل هذه النظم التطبيقية مايلى:

١- نظام خبير تعليمي مبسط للتعرف على الحيوانات.

٢- نظام أولي لتقدير أفضل المسارات بين المدن.

٣- نظام إختيار المسار الآمن.

٤- محاكاة الدوائر المنطقية الرقمية.

٥- حل الالغاز.

(١-١٤) نظام خبير تعليمي مبسط للتعرف على الحيوانات

(Small Expert System for Animal Identification)

لبناء نظام خبير (Expert System) يمكنه التعرف علي مسمي وصفات أحد الحيوانات أو الطيور من خلال البحث داخل قاعدة معارف (Knowlegde Base) تحتوي مثلا علي مسمي وصفات سبعة من الحيوانات والطيور تتخذ الخطوات التالية:

١ - الحوار النهائى: يتم تحديد الشكل العام للحوار النهائى (Dialogue) بين المستخدم (User) والنظام الخبير والذي يمكن أن يأخذ مثلا شكل الاسئلة المتتالية الآتية:

Goal : _ run .

النظام الخبير has it hair ?

المستخدم yes

النظام الخبير does it eat meat ?

المستخدم yes

النظام الخبير has it a fawn color ?

المستخدم yes

النظام الخبير has it dark spots ?

المستخدم yes

النظام الخبير your animal may be a (an) cheetah !

علما بأن النظام الخبير لايقوم بطرح أحد الاسئلة الا إذا تم الاجابة علي السؤال السابق (بنعم أو لا) من قبل المستخدم بإستخدام لوحة المفاتيح ، وبعد السؤال الرابع فان النظام سوف يستدل علي إسم الحيوان.

٢- العلاقات الاسنادية: يتم بناء التوصيف للعلاقات الإسنادية والبندود كما يلى:

أ- التوصيف للعلاقات الإسنادية (predicates) والتي تشمل توصيف إسم الحيوان

(cheetah , tiger , ostrich , ...)

وكذلك نوع الفصيلة (... , bird , mamal) على أنهم رموز، ثم تعرف العلاقة الإسنادية

الايجابية بين أسم الحيوان وأحد صفاته مثل :

positive(has,dark_spots) : لتأخذ الشكل :

poitive(symbol,symbol) : كما هو مبين فيمايلي:

predicates
animal_is(symbol)

it_is(symbol)

positive(symbol,symbol)

ب - التوصيف للتعبيرات أو البنود (clauses) والتي المحتوية على الحقائق (Facts)

clauses

animal_is(cheetah) if
it_is(mammal) and
it_is(carnivore) and
positive(has,tawny_color) and
positive(has,dark_spots) .

animal_is(tiger) if
it_is(mammal) and
it_is(carnivore) and
positive(has,tawny_color) and
positive(has,black_stripes) .

animal_is(giraffe) if
it_is(ungulate) and
positive(has,long_neck) and
positive(has,long_legs) and
positive(has,dark_spots).animal_is(zebra) if
it_is(ungulate) and
positive(has,black_stripes) .

animal_is(ostrich) if
it_is(bird) and
negative(does,fly) and
positive(has,long_neck) and
positive(has,long_legs) and
positive(has,black_and_white_color) .

animal_is(penguin) if
it_is(brid) and
negative(does,fly) and
positive(does,swim) and
positive(has,black_and_white_color) .

animal_is(albatross) is
it_is(brid) and
positive(does,fly_well) .

والقواعد (Rules) التي تأخذ الشكل الآتي:

it_is(mammal) if
positive(has,hair) .

it_is(mammal) if
positive(does,give_milk) .

```

it_is(brid) if
    positive ( has , feathers ) .
it_is(brid) if
    positive(does,fly) and
    positive(does,lay_eggs) .
it_is(carnivore) if
    positive(does,eat_meat) .
it_is(carnivore) if
    positive(has, pointed_teeth) and
    positive(has,claws) and
    positive(has,forward_eyes) .
it_is(ungulate) if
    it_is(mammal) and
    positive(has,hooves) .
it_is(ungulate) if
    it_is(mammal) And positive(does,chew_cud)
    
```

يمكن إضافة تعبيرات أو بنود تحتوي علي حقائق الي قاعدة البيانات، وذلك بإستخدام العلاقة القياسية (asserta) مثل :

```
asserta(xpositive(has,black_strips))
```

والتي تضيف الحقيقة الإيجابية التالية الى قاعدة البيانات (database) :

```
xpositive(has,black_strips)
```

وذلك بشرط أن يجري تعريف العلاقة (xpositive) في قاعدة البيانات في بداية البرنامج والتي تأخذ الشكل :

```

domains
.....
database
    xpositive(symbol,symbol)
predicates
.....
clauses
.....
    
```

مع ملاحظة عدم إدخال القواعد (Rules) في تعريف قاعدة البيانات، وبذلك تأخذ قاعدة البيانات الشكل الآتى:

```

database
    xpositive(symbol,symbol)
    xnegative(symbol,symbol)
    
```

وتكون العلاقة بين positive , xpositive هي القاعدة :

```
positive(X,Y) if xpositive(X,Y) , ! .
```

يمكن إستخدام القاعدة المماثلة للنفي:

```
negative(X,Y) if xnegative(X,Y) , ! .
```

ويمكن إستخدام العلاقة الإسنادية (ask) لتوجيه السؤال الي المستخدم في حالة إذا لم

يجد البرنامج إجابات تعارض القاعدة الموجبة كمايلي:
positive(X,Y) if
not(xnegative(X,Y)) and ask(X,Y,Yes) .

أو تعارض القاعدة التالية كمايلي:
negative(X,Y) if
not(xpositive(X,Y)) and
ask(X,Y,no)

يمكن للعلاقة (ask) ان توجه السؤال وتذكر الاجابة وأستخدم (y) للإيجاب و (n) للنفي كمايلي :

```
ask(X,Y,yes) :-
    write(X,"it",Y," \n") ,
    readln(Reply) ,
    frontchar(Reply,'Y',_) , ! ,
    remember(X,Y,Yes) .
```

```
ask(X,Y,no) :-
    write(X,"it",Y," \n") ,
    readln(Reply) ,
    frontchar(Reply,'n',_) , ! ,
    remember(X,Y,no) .
```

```
remember(X,Y,Yes) :-
    assert(xpositive(X,Y)) .
```

```
remember(X,Y,no) :-
    assert(xnegative(X,Y)) .
```

٣- إ حراز الهدف لإحراز الأهداف فانه يمكن تحميل البرنامج وتنفيذه وذلك بكتابة:

```
run :-
    animal_is(x) , ! ,
    write("\n Your animal may be a(an)",X) ,
    nl,nl,clear_facts .
```

```
run :-
    write("\n Unable to determine what ") ,
    write("your animal is : \n \n"),clear_facts .
```

حيث يبدأ البرنامج في طرح الاسئلة واحداً بعد الآخر وأستقبال الاجابة بنعم أو لا واعطاء

الاجابة الصحيحة في شكل : Your animal may be a(an)

أو الاجابة التي تفيد أن البرنامج غير قادر على تحديد مسمى الحيوان في الشكل:

Unable to determine what your animal is .

يلاحظ إستخدام العلاقة (Clear-facts) وذلك للتخلص من جميع الحقائق التي تم إدخالها بواسطة المستخدم عند الاجابة علي الاسئلة حتي لا تؤثر علي الحقائق المدخلة من قبل في إحراز الهدف السابق إن وجد وإضافه شرط الخروج من البرنامج والرجوع الي نظام

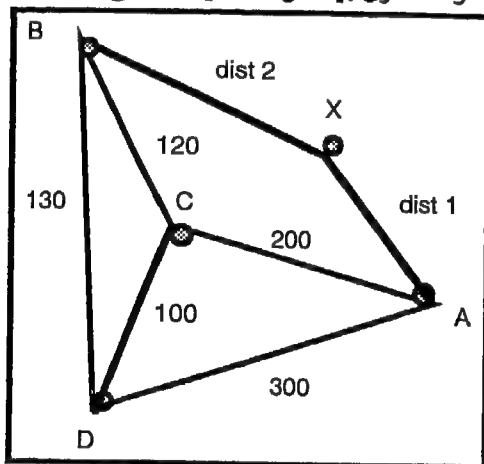
البرمجة المنطقية السريعة بالضغط علي عمود المسافات كمايلي:
clear_facts : _
retract(xpositive(_,_)) , fail .

```
clear_facts : _
    retract(xnegative(,_)) , fail .
clear_facts : _
    write("\n\n please press the space bar to exit \n") ,
    readchar(_)
```

(١٤-٢) نظام أولى لتقدير أفضل المسارات بين المدن

(Prototyping for Best Routing between Cities)

لتصميم برنامج يمكنه البحث عن أفضل المسارات بين المدن، والذي يمكن أن يعتبر نظام نموذج أولى (Prototype) لنظام أكبر يجري إستكماله بعد ذلك. لنفرض ان هناك أربعة مدن (A , B , C , D) مبينة على خريطة ما حيث تبلغ أطوال الطرق بين كل مدينتين كالآتي :



الطريق بين (A) , (C) = ٢٠٠ كيلومترا
 الطريق بين (A) , (D) = ٣٠٠ كيلومترا
 الطريق بين (C) , (D) = ١٠٠ كيلومترا
 الطريق بين (B) , (C) = ١٢٠ كيلومترا
 الطريق بين (B) , (D) = ١٣٠ كيلومترا
 ويراد توجيه الاسئلة المشابهة لمايلي الى النظام:

١- هل هنال طريق يربط مدينه باخري؟

٢- هل هناك مدن تقع عل مسافه اقل من ٥٠ كم ؟

لتصميم ذلك فإن البرنامج يمكن أن يشتمل على الآتي:

أ - المجالات (domains) والتي تحتوى على التعريف الرمزي للمدينة (town) والمسافة (distance) على انهما أرقام صحيحة.

ب - توصيف العلاقات الاسنادية (predicates) حيث توصف العلاقة طريق (road) على انها المسافة بين مدينتين، وأن افضل المسارات (route) تمثل مسافة بين مدينتين.

ج - التعبيرات أو البنود (clauses) والتي تحتوي علي الحقائق التي تمثل المسافات بين المدن المختلفة، وكذلك القواعد التي تأخذ الشكل الآتي:

```
route(Town1,Town2,Distance) : -
    road(Town1,Town2,Distance) .
```

والتي تبين أن أفضل المسارات يمكن أن يكون الطريق بين مدينتين إن وجد أو يمكن أن

يكون علي مرحلتين (Dist 1): من المدينة الاولى الى المدينة (X) ثم (Dist 2): من المدينة (X) الى المدينة الثانية، وأن أفضل المسارات هو مجموع المسافتين كمايلي:

```
route(Town1,Town2,Distance) :-
    road(Town1,Town2,Distance) .
route(Town1,Town2,Distance) :-
    road(Town1,X ,Dist1) ,
    route(X,Town2,Dist2) ,
    Distance=Dist1+Dist2,!.

```

فيمايلي البرنامج الكلى:

```
domains
    town = symbol
    distance = integer
predicates
    road(town,town,distance)
    route(town,town,distance)
clauses
    road(A,C,200) .
    road(D,A,300) .
    road(C,D,100)
    road(C,B,120) .
    road(D,kansas_city,140) .
    route(Town1,Town2,Distance) :-
    road(Town1,Town2,Distance) .
    route(Town1,Town2,Distance) :-
    road(Town1,x,Dist1) ,
    route(x,Town2,Dist2) ,
    Distance=Dist1+Dist2,!.

```

يمكن إحراز الاهداف التالية: route(C,B) أو route(A,B,X) أو ما شابه ذلك .

(١٤-٣) (نظم الانقاذ) اختيار المسار الآمن

(Safe Routing Selection System)

لإبراز قدرة لغة البرمجة المنطقية السريعة (TP) علي بناء برامج ونظم للذكاء الاصطناعي

فاننا نورد المثال التالي: نفرض ان هناك خريطة لجبل ما شكل (١٤-١) الذي يحتوى على عدة

مغارات فى شكل حجرات تحتوى ثلاثة منها علي الأخطار الآتية :

١- الخطر الأول (danger 1)

٢- الخطر الثانى (danger 2)

٣- الخطر الثالث (danger 3)

وتحتوى ثلاثة اخرى على مايلي:

١- حجرة التزود بالطعام (Food)

٢- حجرة التزود بالمياه (water)

٣- حجرة للراحة (rest)

وتحتوى الغرفة السابعة على مجموعة من الأطفال (children). والمطلوب تصميم برنامج يسمح بالدخول من المدخل وإنقاذ الأطفال والخروج بهم من المخرج مع تجنب الاخطار. ويجري بناء البرنامج لهذه المشكلة كمايلي:

١- المجالات (domains) والتي تشتمل على تعريف الحجرة (room) علي انها رمز وقائمة الحجرات (roomlist) علي انها قائمة تضم عددا من الحجرات والتي تبدأ خالية (room*) كمايلي:

domains

room = symbol

roomlist = room *

٢- التوصيف للعلاقات الإسنادية (predicates) حيث يجري توصيف العلاقة مر (gallery) على انه يربط بين حجرتين، والعلاقة حجرة مجاورة (neighborroom) على انها تربط بين حجرتين بدون النظر الى إتجاه التحرك في الممر، وان المسار (route) هو الذي يحدد علاقة البداية من حجرة تم زيارتها الى حجرة أخرى لم يتم زيارتها مع وضع جميع الحجرات التي تم زيارتها فى القائمة (roomlist) كما هو مبين :

predicates

gallery(room,room) /* there is a gallery between two rooms */

neighborroom(room,room) / * Necessary because it does not matter which direction we go along a gallery */

avoid(roomlist)

go(room,room)

route(room,room,roomlist) / * this is the route to be followed . Roomlist consists of a list of rooms already visited . */

member(room,roomlist)

٣- التعبيرات (clauses) والتي تتكون من الآتى :

١- الحقائق والتي تشمل نقل حقائق الخريطة فى شكل ممرات بين كل حجرتين كمايلي:

clauses

gallery(entry,danger 1) .

gallery(rest,danger 2).

gallery(exit,children).

gallery(danger 3,children).

gallery(food,children).

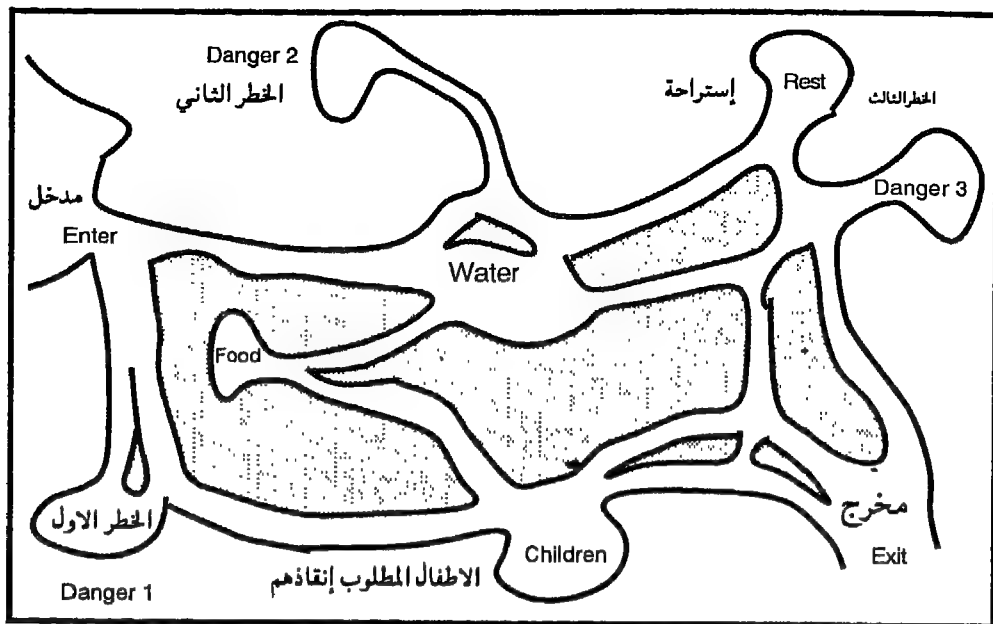
gallery(entry,rest) .

gallery(rest,food) .

gallery(rest,water) .

gallery(rest,danger 3) .

gallery(water,exit) .



شكل (١٤-١)

gallery(danger 1,children) .

٢- القواعد والتي تشتمل على الشروط التي تحدد الآتى:

- أ - ان الحجرتين (X,Y) تكون متجاورتين اذا كان هناك ممر يربط بينهما .
- ب - أمر الذهاب الي هنا او هناك بشرط ان يتم الذهاب الي هنا اولاً .
- ج - تحديد مسار الخروج بشرط ان يتم زيارة الحجرة التي بها الاطفال مع كتابة اسم الحجرة وتجنب الحجرات المحتوية علي الاخطار كما هو مبين بالآتى:

```
neighborroom (X,Y) if gallery(X, Y) .
neighborroom(X,Y) if gallery(X,Y) .
avoid([ danger 1 , danger 3]) .
go(Here,Tere) if route(Here,There, [Here]) .
route(exit,exit,VisitedRooms) if
    member(children, VisitedRooms) and
    write(VisitedRooms) and nl
route(Room,way_out ,VisitedRooms) if
    neighborroom(Room,Nextroom) and
    avoid(DangerousRooms) and
    not(member(NextRoom,DangerousRooms)) and
    not(member(NextRoom,VisittedRooms)) and
    route(NextRoom,way_out,[ Nex,|VisitedRooms]) .
member(X,[ X | _]) .
member(X,[ _ | H]) if member(X,H) .
```

٤- عند طلب احراز هدف فى شكل go(entry,exit) فإن البرنامج سوف يكتب أول محاولة

ناجحة للخروج بعد ان ينقذ الاطفال، ويمكن إستخدام العلاقة (fail) لإجبار البرنامج على التتبع الخلفى بعد أن يصل الى أول محاولة ناجحة كمايلي:

```

rout(Room,Room,visitedRooms) if
member(children,visitedRooms) and
write(visitedRooms) and nl and
fail .

```

وبذلك نحصل علي جميع المحاولات الناجحة لتحديد المسار الآمن لإنقاذ الاطفال.

(١٤-٤) محاكاة الدوائر المنطقية الرقمية

(Simulation of Digital Logic Circuits)

كما تم ذكره سابقا من اعتماد لغة البرمجة المنطقية السريعة على العلاقات الإسنادية الأساسية التى توصف الدوائر المنطقية مثل (XOR,NOT,OR,AND)، والتى يمكن ان تستخدم فى بناء دوائر منطقية اكبر حجما. ويبين شكل (١٤-٢) دائرة منطقية رقمية تقوم بعمل دائرة (XOR) ويمكن تحويل هذه الدائرة الي برنامج يحتوي على:

١- المجالات (domains)
domains
d = integer

والتي تبين تعريف مجال الحرف على أنه عدد صحيح.

٢ - الإسنادات (predicates) والتي تحتوى على تعريف للعلاقات الإسنادية لكل دائرة منطقية منفصلة والذي يشمل الربط المنطقى بين المدخل والمخرج.

```

predicates
not_(D,D)
and_(D,D,D)
or_(D,D,D)
xor_(D,D,D)

```

٣ - البنود (clauses) والتي تحتوى على:

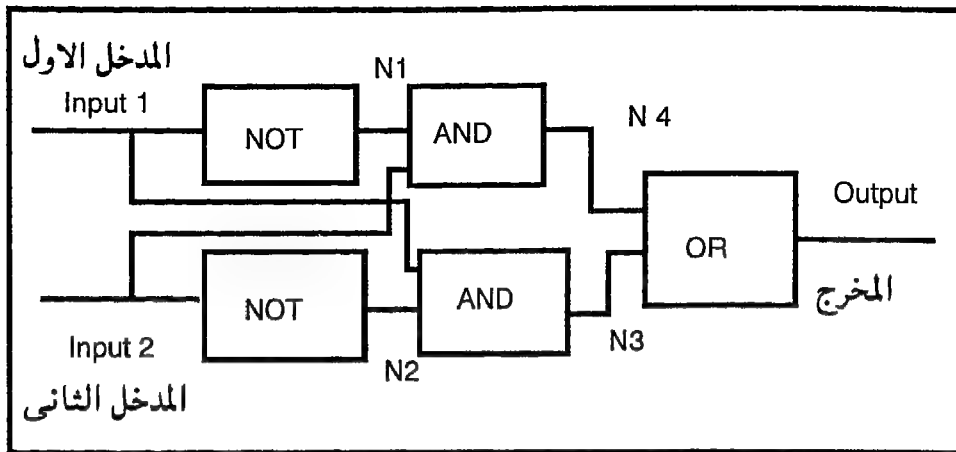
أ- الحقائق المنطقية لإحتمالات المدخل والمخرج لكل دائرة منطقية على حدة.

```

clauses
not_(1,0) . not_(0,1) .
and_(0,0,0).and_(0,1,0).and_(1,0,0).and_(1,1,1) .
or_(0,0,0).or_(0,1,1).or_(1,0,1). or_(1,1,1) .

```

ب - القواعد الشرطية للدائرة والتي تفيد ان هذه الدائرة سوف تمثل (XOR) اذا شملت دائرتين (NOT) ودائرتين (AND) ودائرة ثالثة (OR) بالعلاقة الإسنادية الموصفة للمدخل والمخرج لكل منها كمايلي:



شكل (١٤-٢) دائرة او المانعة (XOR)

```
xor( input1 , input 2 , output ) if
not_ ( input 1,N1) and not_(Input 2 , N 2) and
and_ ( Input 1, N 2, N 3) and not_( Input 2,N 1,N 4) and
or_( N 3, N4, 0 ) .
```

٤ - يتم إحراز الهدف التالى:

xor(Input1,Input2,Output)

و سوف يجيب البرنامج بجدول الحقيقة (Truth-Table) كمايلي:

Input1 = 0 , Input 2 = 0 , output = 0

Input1 = 0 , Input 2 = 1 , output = 1

Input1 = 1 , Input 2 = 0 , output = 1

Input1 = 1 , Input 2 = 1 , Output = 0

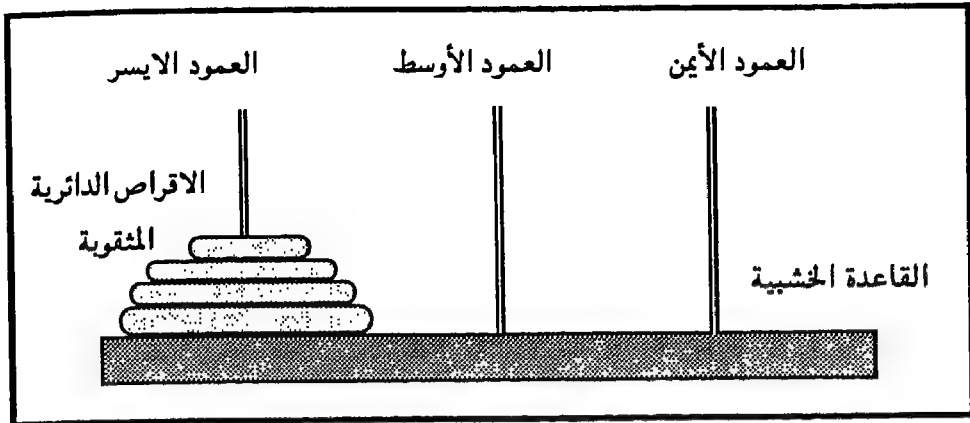
4 solutions.

(١٤-٥) حل الألغاز (Puzzle Solving)

لبيان كيف يمكن إستخدام البرمجة المنطقية السريعة في إيجاد الحلول للألغاز فاننا نورد المثال التالى: يتركب لغز " أبراج هانوى " من ثلاثة أعمدة (الأيمن والأوسط والأيسر) مركبة على قاعدة خشبية، ويحمل العمود الأيسر عددا من الاقراص الخشبية الدائرية الشكل (N) والمثقوبة من المنتصف والمتتالية الاقطار والمرتبة ترتيبا تنازليا طبقا للأقطار كما هو مبين بالشكل (١٤-٣) ، يراد نقل هذه الاقراص من العمود الايسر الي العمود الايمن بإستخدام العمود الاوسط كعامل مساعد بحيث يتم النقل في النهاية بنفس الترتيب مع تنفيذ الشرط :

" أن لا يكون هناك فى أى حال من الأحوال أثناء النقل أن يقع قرص كبير فوق قرص أصغر".

ويكون حل اللغز سهلا اذا كان عدد الاقراص واحدا او اثنين، ولكن اذا زاد العدد عن ذلك



شكل (١٤-٣) لغز " أبراج هانوي "

فإن الحل يصبح معقداً. يمكن ان تتبع الإستراتيجية الآتية فى بناء البرنامج اذا كان عدد الأقراص (N) :

- ١- نقل الأقراص (N-1) من العمود الأيسر الى العمود الاوسط .
 - ٢- نقل القرص الأخير من العمود الأيسر الى العمود الأيمن.
 - ٣- نقل الأقراص (N-1) من العمود الأوسط الى العمود الايمن .
- بذلك يصبح البرنامج فى الشكل التالى:

domains

pos = right ; middle ; left

predicates

tower(integer)

move(integer, pos,pos, pos)

inform(pos, pos)

clauses

tower(N) : - move(N,left,middle,right) .

move(1,A,C) : - inform(A,C) , ! .

move(N,A,B,C) : -

N1= N-1,move(N1, A,C, B),inform(A,C),move(N1,B, A,C) .

inform(pos1, pos2) : -

write(" \ nMove a disk from ", pos1, " to ", pos2) .

tower (3)

ويطلب إحراز الهدف التالى:

فإن البرنامج سوف يكتب الآتى:

Move a disk from left to right
Move a disk from left to middle
Move a disk from right to middle
Move a disk from left to right
Move a disk from left to right
Move a disk from middle to left
Move a disk from middle to right

Move a disk left to right

(١٤-٦) تخمين الكلمات (Word Guessing)

كما يبين البرنامج التالي كيف يمكن تخمين الكلمات (Word Guessing) وذلك بإختيار كلمة مجهولة للاعب، ويتسلسل البرنامج كما في الخطوات التالية:

- ١- تعريف عدد الأحرف المكونة للكلمة ، ويتم ادخال ذلك في البرنامج $\text{word}(w, L)$
- ٢- يخمن اللاعب أحد حروف الكلمة، ويتم ادخاله من لوحة المفاتيح فإذا كان هذا الحرف من مكونات الكلمة فانها تذهب الى النافذة الایجابیة - نافذة نعم - وإذا لم يكن هذا الحرف من مكونات الكلمة فإنه يذهب الى نافذة النفي - نافذة لا .
- ٣- يتولي تخمين الحروف الاخرى المكونة للكلمة ويتوالي كتابتها في احد النافذتين الى ان يتم الوصول الى تخمين صحيح للكلمة.
- ٤- يقوم البرنامج بحساب نسبة الخطأ والصواب للاعب .

domains

list = symbol*
scores = integer

predicates

member(symbol , list)
run continue(list , scores)
yes_no_count(symbol , list)
guessword(scores , list)
word(list , integer)
read_as_list(list , integer)
goal makewindow(1 , 7 , 0 , " " , 0 , 0,25, 80) ,
makewindow(2,7, 145, " counting" , 1, 20 , 4 , 34) ,
makewindow(3, 112 , 112 , " YES " , 5 , 5 , 7 , 30) ,
makewindow(4 , 112 , 112 , " N O " , 5 , 40 , 7 , 30) ,
makewindow(5 , 7 , 7 , " " , 14, 20 , 10 , 34) ,
run .

clauses

run : -

word(W,L) , shiftwindow(1) , clearwindow ,
write(" The word has " , L , : letters ") ,
shiftwindows(2) , clearwindow ,
shiftwindows(3) , clearwindow ,
shiftwindow(4) , continue (W , 0) , fail .
continue (L , R) : -
shiftwindow(5) , clearwindow ,
write(" Guess a letter : ") ,
Total = R + 1 , readln (T) _ , yes _ no _ count) T , L) ,
shiftwindow(5) , clearwindow ,
guessword(Total , L) , continue (L , Total) .

```

yes_no_count( X , List ) :-
    member( X , List ) , shiftwindow( 3 ) , write( X ) ,
    shiftwindow( 2 ) , write( X ) , ! .
yes_no_count( X , _ ) :-
    shiftwindow( 4 ) , write( X ) ,
    shiftwindow( 2 ) , write( X ) .
guessword( Count , Word ) : _
write( " Know the word yet ? press y or n " ) ,
readchar( A ) , A = ' Y ' , cursor( 0 , 0 ) ,
    write("Type it in one letter per line \n " ) ,
    word(word,L),read_as_list(G,L),
    G = word,clearwindow,window_attr(112) ,
    write("Right ! you used",count,"guess(es)" ) ,
readchar( _ ) , window_attr(7) , ! , fail .
guessword( _ , _ ) .
word([b,i,r,d],4).word([p,r,o,l,o,g],6) .
word([f,u,t,u,r,e],6) .
member(X,[X|_]):- ! .
member(X,[_|_]): -member(X,T) .
read_as_list([ ],0):- ! .
read_as_list( [Ch|Rest],L ) :-
    readln(Ch) , L1=L-1 , read_as_list(Rest,L1) .

```

الفصل الخامس عشر

لغة البرمجة

باسلوب القائمة

LISP

(١٥ - ١) التعرف على لغة البرمجة بالقائمة (List Processing)

يمكن إطلاق اسم لغة البرمجة بإسلوب القائمة علي هذه اللغة، ويمكن للإنسان لأول وهلة التعرف علي برامج هذه اللغة من الشكل العام للكتابة، حيث تحتوي كل جملة علي قوس كامل علي الأقل كما هو واضح في الامثلة المبينة فيما يلي، حيث توجد نجمة قبل ذلك القوس (Asteris) والتي تعنى إيجاد القيمة (eval) لما يتم كتابته في داخل القوسين، ووجود مسافة فارغة علي الاقل بين مفردات عناصر الجملة. يمكن كذلك ملاحظة ان جملة لإيجاد الجذر التربيعي مثلاً للرقم (٣٦) تنفذ فوراً عندما يكتب الطرف الثاني من القوس، وأنه يمكن كتابة الارقام بدون قوس، وأن غياب الطرف الثاني من القوس يمنع تنفيذ هذه الجملة كما هو واضح في الامثلة التالية :

* (sqrt 36)

6

*9

9

*()

nil

ومع أن اللغة تعتبر لغة غير حسابية فإنه يتم تنفيذ العمليات الحسابية لما بين القوسين فوراً مثل إيجاد الفرق أو إيجاد القيمة المطلقة (abs) أو وضع الارقام في وضع تبادلي. توضع الأمثلة

* (min 7 4 9 5 7)	*(* 4 5 6)
4	120
* (-4.98 6.38)	*(max 3 5 9 12 1 0 8)
1.40	12
*(+1 2 3)	*(exp 1)
6	2.7182818
* (abs -9.50)	* (= 6 (/60 10)
9.50	t
*(expt 5 2)	*(> 9 3)
25	t
* (quote(Mohamed Ahmed.))	*(>= 7 6 _
Mohamed Ahmed.	t
* (List U .A.R)	*(> 6 6)
(UAR)	nil

التالية جملة لايجاد القيمة الصغرى (min) لبعض الأرقام، والذي يقابله برنامج مكون من أكثر من جملة فى اللغات الاخرى وكذلك ايجاد القيمة المطلقة والرفع الى الاس والجمع والطرح كما فى الجدول المبين.

(١٥-١-١) الذرات (Atoms) والقوائم (Lists)

من أهم سمات لغة البرمجة بالقائمة أنها لغة رمزية (Symbolic) تأخذ فيها التعليمات (Instructions) للبيانات نفس التعبير الرمزي الشكل لتقارب اللغة الطبيعية (Natural Language) وأنها تتكون من ذرات (Atoms) بمعناها الشكل العام المجازي من انها أصغر شيء لا يمكن تفتيته وانها ليست قائمة، وعلي ذلك فإن الأرقام الصحيحة (Integers) والاخرى ذات النقطة العشرية المتحركة (Floating) والتي تقبل القسمة تعرف على أنها ذرات رقمية (Numerical Atoms) كما يجرى تعريف الأشكال مثل حرف (T) أو أسماء العمليات الرياضية مثل اجراء الجمع (Sum)، والمجموع الكلى (Total) أو الافعال أو الاسماء العادية مثل رجل او طفل و ما الى ذلك على أنها ذرات شكلية او رمزية (Atoms Symbolic).

يجرى تعريف القائمة (List) بأنها الجملة التى تبدأ بالطرف الأيسر للقوس ويمكن ان تشتمل على أصفار او ذرات او قوائم كاملة اخرى كما هو واضح فى القوائم التالية :

- 1- (this is a list)
- 2- ((this is a list) within a list)
- 3- (defun both -empty (a b) (an (null a) (null b)))

يعتبر العنصر الاول فى القائمة الذى يلى بداية القوس هو الدالة (Function) التى يجرى تنفيذها على بقية العناصر التالية التى تعتبر المدلول للعملية (Argument) مثال :

*(max 3 5 9 12 1 0 8)
12

(١٥-٢) تراكيب البرامج (Programming Structure)

يتركب البرنامج المكتوب بلغة البرمجة بأسلوب القائمة من جزئين أساسيين:

١- الجزء الاول : وهو النموذج (Form) ويعرف بأنه الشكل النمطي الذي يمكن ايجاد قيمة له عند تنفيذه. وتنقسم الاشكال النمطية للنموذج الى خمسة انواع كالآتي :

أ - النموذج الذي يقوم بإيجاد قيمته بنفسه (Self Evaluating) والذي يشتمل على جميع الأرقام والحروف والكلمات ... الخ ..

ب - النموذج المتغير (Variable Form): والذي يمثل بشكل من الاشكال النمطية

block catch copmpiler - let declare eval - when flet function go	if labels let let * macrolet multiple - value - call multiple - value - prog1 progn	progv quate return-from setq tagbody the throw unwhnd-protect
Names Of All Common LISP Special Forms		

شكل (١٥-١) - النماذج الخاصة والمحجوزة للغة البرمجة بالقائمة

ويأخذ قيما متغيرة.

ج - النموذج الخاص (Special Form) وهو بعض النماذج التي تم حجزها لهذه اللغة، والتي يمكن التعرف عليها للقيام بمهام معينة لكل واحدة علي حدة كما هو مبين في شكل (١٥-١) .

د - الاستدعاء العام (Macro Call) والذي يتمثل بقائمة يكون عنصرها الاول شكلا رمزيا او دالة من غير النماذج المحجوزة في الجدول شكل (١٥-١) والتي تعرف بعد ذلك ويمكن استدعاؤها لتنفيذ ذلك.

هـ - الاستدعاء الدالي (Function Call) هي القائمة التي يكون عنصرها الاول ليس من النوع الخاص او نوع الاستدعاء العام.

٢- الجزء الثاني:

هو الدالة التي تعمل على إيجاد القيمة للمدلول. ولقد تم البدء باستخدام الارقام وذلك لتشجيع القاريء والمبرمج العادي بالدخول في تفاصيل البرمجة بالقائمة مع ان العمليات الحسابية ليست هي الهدف الاساسي لهذه اللغة. وتكمن قوة هذه اللغة في انها تقوم بتحليل والمعالجة للنمط الشكلي او الرمزى (Symbolic Manipulation).

(٣-١٥) العمليات الحسابية المنطقية (Arithematic Manipulation)

تعتبر القائمة الصفرية (nil) والصفر (null) والحرف (T) من الثوابت في لغة البرمجة بالقائمة، ويستخدم الصفر لتوقع نتيجة منطقية غير صحيحة (False) ويرمز لها بالرمز (F) أما إذا كانت النتيجة صحيحة (True) فيرمز لها بالرمز (T) وذلك عند تنفيذ الدوال الحسابية مثل التساوى وأكبر من وأقل من وإيجاد النهايات العظمى والصغرى. مثال :

* (setq x 5) 5	* (setq x (+2 3)) 5
* (setq y 7) 7	* (setq Hometown ' (Cairo)) (Cairo)
* (psetq x y y x) *x 7 *y 5	* (Setq) nil * (setq x (-9 4) y x) 5

شكل (١٥-٢) دوال تخصيص القيم

*(null nil)
t

*(null (a b))
nil

(١٥-٣-١) التعامل بالقوائم (List Operations)

من أهم مميزات لغة البرمجة بالقوائم أنه يوجد كثير من الدوال مثل (Set) , (List) (Append) التى تتعامل مع القوائم والتى توضح ان هذه اللغة شديدة القرب من اللغة الطبيعية، ومن أهم الاستخدامات يمكن تلخيص الآتى:

أ - دوال تخصيص القيم (Assign Values) :

وهى التى تستخدم لتخصيص قيم مثل (Set , Setq , Pset) ولتوضيح إمكانية التعامل مع القوائم فإن الدالة (Setq) عند تنفيذها يأخذ العنصر الاول قيمة العنصر الثانى كما هو واضح فى الامر (Set) كما هو واضح فى الجدول (١٥-٢).

ب - دوال تجزئة القوائم الى أجزاء (Take List Apparts) :

وهى التى تجزئ القائمة الى أجزاء مثل الدالة (Car) التى تختار العنصر الاول فقط (أى رأس القائمة) فقط و الدالة (Second) التى تأخذ العنصر الثانى فقط، ثم الدالة (Cdr) التى تختار كل عناصر القائمة ماعدا العنصر الاول أى أنها تختار ذيل القائمة، وهذا واضح فى شكل (١٥-٣) .

ج - دوال بناء القوائم وعرضها (Construction&Display)

من أهم دوال بناء القوائم من عناصرها هى الدوال (List) , (Append) , (Cons) وهى التى تقوم بتجميع عناصر القوائم بترتيب مختلف فى كل حالة لتعطى قائمة خاصة طبقاً لنوع الدالة المستخدمة كما هو واضح فى شكل (١٥-٤) وشكل (١٥-٥) .

د - دوال إعادة التنظيم (Reorganise List)

هذه الدوال مثل (Member) التي تعمل على التحقق من وجود اسم داخل قائمة او التأكد

```
* ( car ' ( x y z ))
x
* ( car ' (( USA Washington - DC ) China ))
( USA Washington - DC )

* ( car ( car ( cdr ' (( plus 7 8 ) ( x y ) 1 )))
x

* ( cdr ' ( x y z ))
(y z)

* ( second ' ( a b c d ))
b
```

شكل (١٥-٣) دوال تجزئة القوائم الى رأس القائمة وذيلها

<pre>*(atom 'expert) t *(listp '(expert system)) t *(listp 'expert) nil *(setq p 'x) x *(eq p 'x) x *(eq p q) nil *(eq p (car (list p q))) t *(eq (float 3) (float 3)) nil</pre>	<pre>*(equal 5 (- 7 2)) t *(setq x '(p q)) (p q) *(list 'U 'S 'A) (U S A) *(Cons X X X) ((p q) p q p q) *(append X X X) (p q p q p q) *(cons (car X) (cdr X)) X</pre>
--	---

شكل (١٥-٤) - دوال إعادة بناء القوائم وعرضها بشكل آخر.

من وجود حرف داخل كلمة حيث يجرى تمثيل الكلمة كقائمة مكونة من أحرف مثلاً، وكذلك الدالة إتحاد (Union) التي تعمل على إتحاد مجموعة عناصر لأكثر من قائمة وتكون النتيجة قائمة واحدة متحدة العناصر. ودالة التقاطع (Intersection) التي تعطي العنصر المشترك بين أكثر من قائمة، وكذلك دالة إيجاد الفرق أو طرح القوائم (Set difference) التي تقوم بطرح قائمة من أخرى، ودالة إيجاد التكرار (Length) والتي تقوم مقام العداد في اللغات الأخرى، وهذا واضح من الشكل (١٥-٦) الآتي:

```

* ( apply # ' cons ' ( because ( I Love Summing )))
( because I Love Summing )

* ( setq fastly '( David Lisa Eileen Emily))
(David Lisa Eileen Emily )
* ( and ( member ' Lisa family ) ( member ' Tam family ))
( Lisa Eileen Emily )

* ( mapcar # ' + ' ( 1 5 10 20 25 ) '( 1 2 3 4 5 ))
( 2 7 13 24 30 )

```

شكل (١٥-٥) - مثال آخر لدوال إعادة البناء والعرض والتجميع

```

*(setq Love '(I Love Summing ))
( I Love Summing )
*(reverse Love )
( Summing Love I )
*(reverse 'p '(p q r p s))
(q r s )
* ( member Dave ' (peter Lin Dave Linda ))
(Dave Linda )
*(union '(peter Lin Dave Linda ) '(Dave Lisa Eileen Emily ))
(peter Lin Dave Linda Lisa Eileen Emily )
*(intersection '(peter Lin DAve Linda ) '(Dave Lisa Eileen Emily ))
(Dave )
*(setdifference '(peter Lin Dave Linda ) '(Dave Lisa Eileen Emily )
(peter Lin Linda )
*(list '(peter Lin DAve Linda ))
(Linda )
*(length '(p q )0
nil
*( length '((p q) (p q) (p q)))
3

```

شكل (١٥-٦) - دوال إعادة التنظيم

(١٥-٤) الدوال الأكثر استخداماً

(Commonly Used LISP functions)

سوف نتناول عرض الدوال الأكثر استخداماً في الأنماط الأكثر شيوعاً للمستخدم وهي فرائس

ليسب (FranzLISP) والليسب العام (CommonLISP) والمبينة فيما يلي:

١- الدوال الإسنادية (Predicat Functions)

وتعطى هذه الدوال الاسنادية اما صحيح (t) أو الصفر (nil).

القيمة المعطاة	الاستخدام	الدالة الاسنادية
t iff arg is not a list تكون النتيجة صحيحة اذا كان المدلول عليه عنصر وليس قائمة	(atom arg) تؤكد الدالة ان المدلول عليه عنصر وليس قائمة	atom
t iff arg1 and arg 2 are equal تكون النتيجة صحيحة اذا كانت القيم متساوية	(equal arg1 arg2) تؤكد الدالة ان المدلولين عليهما متساويين في القيمة	equal

٢- الإسنادات الحسابية (Arithmetic Predicates)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
t if arg1 = arg2 تكون النتيجة صحيحة اذا كانت القيمتين متساويتين	(= arg1 arg2) تؤكد الدالة التساوي بين قيمتين	=
t if arg1 > arg2 تكون النتيجة صحيحة اذا كانت القيمة الاولى اكبر من الثانية	(> arg1 arg2) تؤكد الدالة القيمة الاولى اكبر من الثانية	>
t if arg1 < arg2 تكون النتيجة صحيحة اذا كانت القيمة الاولى اصغر من الثانية	(< arg1 arg2) تؤكد الدالة القيمة الاولى اصغر من الثانية	<
t if arg1 > or = arg2 تكون النتيجة صحيحة اذا كانت القيمة الاولى اكبر من او تساوي الثانية	(>= arg1 arg2) تؤكد الدالة القيمة الاولى اكبر من او تساوي الثانية	>=

٣- التعامل بالقوائم (List Manipulation)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
the first element of Large:L تكون النتيجة الحرف الاول من الكلمة	(car L arg) تقوم الدالة بتحديد الحرف الاول من الكلمة (الرأس)	car

القيمة المعطاة	الاستخدام	الدالة الاسنادية
the list Large without the first element : arg تكون النتيجة باقي الحروف (الذيل) ماعدا الحرف الاول من الكلمة	(cdr L arg) تقوم الدالة بتحديد باقي الحروف (الذيل) ماعدا الحرف الاول من الكلمة	cdr
a list whose car is arg1 and whose cdr is arg2. (cons abd yrf) abdyrf تكون النتيجة قائمة واحدة	(cons arg1 arg2) تقوم الدالة بضم قائمتين متتاليتين	cons
a list containing the elements of Larg1 followed by Large 2 (append abcd 12567890) abcd12567890	(append Large1 Larfe2) تقوم الدالة بضم قائمتين متتاليتين	append
تكون النتيجة عدد العناصر في كل جزء من القائمة the number of elements in a list -> one -list ((a b) (c) (def) (gh)) -> (mapcar 'length one -list) (2 1 3 2)	(length a_list) تقوم الدالة بتحديد عدد العناصر في قائمة	length
that part of Larg beginning with the first occurrence of arg1. arg تكون النتيجة العناصر المشتركة	(member arg1 Larg) تقوم الدالة بايجاد العناصر المشتركة في قائمتين متتاليتين	member
the nth element of a list (nth 8 abcdefghijk) تكون النتيجة العنصر المحدد الترتيب	(nth index list) تقوم الدالة بايجاد العنصر ذو الرقم المحدد الترتيب في القائمة	nth
(delete mohamed mohamed aly) aly تكون النتيجة بحذف العنصر المحدد	(delete element list) تقوم الدالة بحذف العنصر المبين من القائمة (delete element list)	delete
(remove mohamed mohamed aly) aly تكون النتيجة بحذف العنصر المحدد	تقوم الدالة بحذف العنصر المبين من القائمة	remove

٤- دوال التخصيص (Assignment)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
each symb i is set to have the value i -> (setq bob 'gray) gray تكون النتيجة القيمة المعينة للرمز	(setq symb1 val i [symb2 val2]) تقوم الدالة بتخصيص الرمز بالقيمة التي تليه	setq
->(set price 4) 4 تكون النتيجة القيمة المعينة للتعبير	(set expr value) تقوم الدالة بتخصيص التعبير بالقيمة التي تليه	setf

٥- دوال الخواص والمشاركة (Property and Association)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
the frist elemt of Larg whose car = arg ->(assos arg Larg) L تكون النتيجة الرأس الغير مشترك	(assos arg Larg) تقوم الدالة بحذف الذيل المشترك	assos
the value associated with prop-erty value of symb تكون النتيجة القيمة المعينة للخاصية	(get symb prop_val) تقوم الدالة بتخصيص الخاصية بالقيمة	get
adds to the property list of name the value under indicator تكون النتيجة القيمة المعينة للخاصية	(putprop name val ind) تقوم الدالة بإضافة القيمة المعينة الي قائمة اسماء الخواص	putprop

٦- دوال حسابية (Arithmetic Functions)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
98 تكون النتيجة القيمة العظمى	(max 3 7 9 0 56 45 98 12) تقوم الدالة بإيجاد القيمة العظمى	max

القيمة المعطاة	الاستخدام	الدالة الاسنادية
0 تكون النتيجة القيمة الصغرى	(min 3 7 9 0 56 45 98 12) تقوم الدالة بإيجاد القيمة الصغرى	min

٧-دوال التعريف والتطبيق (Definition and Application)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
fname تكون النتيجة اسم التعبير	(defun fname arg1 exp) تقوم الدالة بتحديد الاسم لتعبير	defun
func_name تكون النتيجة اسم الدالة المطبقة	(putd alt_name func_name) تقوم الدالة بإيجاد اسم الدالة المطبق	putd
list of the values returned from the functional application -> mapcar 'atom '(a b '(c a) (d) e (fg))) (t t nil nil t .nil) تكون النتيجة صحيحة في حالة التطابق وغير صحيحة في الحالات الاخرى	(mapcar fun Larg1 Larg2 Largn) تقوم الدالة بالتحقق من تطبيقها على القوائم	mapcar
-> (apply 'plus '(1234) 10) تكون النتيجة جمع الارقام في القائمة	(apply fun args) تقوم بتنفيذ عمل الدالة على القائمة	apply

٨-دوال التحكم (Control Functions)

القيمة المعطاة	الاستخدام	الدالة الاسنادية
(cond 1 clause1 1 clausen) A branching function تقوم بتنفيذ التفرع	(cond 1 clause1 1 clausen) A branching function تقوم بتنفيذ التفرع	cond
value of last expression in body exp. تكون النتيجة قيمة المتغير الاخير في جسم القائمة	(let var_binds body_exp) تقوم الدالة بربط القيمة للتعبير الاخير في جسم قائمة	let

القيمة المعطاة	الاستخدام	الدالة الاسنادية
-> (do 'plus '(1234) 10 تكون النتيجة جمع الارقام في القائمة	(do) تقوم بتنفيذ عمل الدالة علي القائمة	do
b if a evaluates to non - nil , els c تكون النتيجة القيمة الاولى اذا كان الشرط لا يساوي الصفر والقيمة الثانية في غير ذلك	(if a b c) تقوم الدالة بتنفيذ التضمين الشرطي	if

(Recursive List reverser) (١٥-٥) التكرار عند عكس عناصر القائمة

يعتبر استخدام التكرار في لغة ليسب عنصراً أساسياً وذلك لسهولة الاستخدام وللطبيعة الرمزية (Symbolic) للبيانات، ويمكن القول بأنه سوف يتم الاستعانة بالتكرار في كثير من الحالات وخصوصاً عند ايجاد وترتيب القوائم لتأخذ الشكل الهرمي (Hierarchical List). لتوضيح ذلك فإننا اذا طلبنا تنفيذ دالة عكس عناصر القائمة في الشكل الآتي :

->rev '(a b c d))

(d c b a) والتي تتولى عكس عناصر القائمة لتعطي القائمة الآتية:

وباستخدام دالة التتبع يمكن الحصول على الخطوات التالية :

```
1- rev [list : (a b c d)]
2- rev [list : ( b c d)]
3- rev [list : ( c d)]
4- rev [list : ( d)]
5- rev [list : (nil)]
5- rev=(nil)
4- rev=(d)
3- rev=(d c)
2- rev=(d cb)
1- rev=(d c b a)
(d c b a)
->
```

والتي تبين ان القائمة تم تجزئتها الى الرأس (car) والذي يشتمل على العنصر (a) والذي (cdr) والذي يشمل (b c d) ثم تكرر هذه الخطوة وفصل الحرف (b) كرأس للقائمة ثم التكرار الى ان نصل الى قائمة صفرية ثم عكس هذه العناصر ويلاحظ ان استخدام التكرار والضم للحصول على قائمة تحتوي على جميع عناصر القائمة الاولى بشكل معكوس.

يمكن كتابة دالة العكس بالشكل الاتي:

```
(defun rev (list)
  (cond
    ((null list) nil)
    (t (append (rev (cdr list))
                (list (car list)))))
```

ويمكن القول بأن دالة العكس (reverse) تفصل وتكرر العناصر من الذيل الى ان تنتهي عند اول العناصر، ولذلك تعرف بأنها دالة التكرار البادئ من الذيل (Tail Recursive).

(١٥-٦) التصنيف والتوحيد للنماذج (Classification / Mode Unification)

اذا تصورنا اننا سوف نحلل احدى الصور الغير ملونة (Grey-Level) ذات الدرجات المتدرجة من الابيض والاسود، وانه قد طلب من ان نبني نظاما يمكنه الاجابة على السؤال الاتي:

هل يوجد بالصورة سيارة او دبابة ؟

(Is there an auto or tank in the image?)

للإجابة على هذا السؤال فإننا نحتاج الى مايلي:

- ١ - خوازميات لاستخراج الملامح الخافتة (Low-level features) من الصورة مثل الحواف (Edges) والتقطيع للشرائح المحددة بخطوط (Line segment) والمنطقية (Regions) وهكذا ووضعها في شكل نموذج مناسب للتعرف والتفرقة بين السيارة والدبابة مثلا.
- ٢ - وضع البرنامج في شكل شبيه باللغة الانجليزية (English-like) ويمكن للحاسب التعامل معه، وبذلك نبدأ بوضع الوصف الآتي للسيارة :

auto

has-part	wheels
has-id	owner
has-location	exterior
has-part	roof
has-for propulsion	engine

wheel

has-part	tire
may -have -part	spokes
may -have -part	fancy -hub -cap
is	round
	unless is flat

and
roof

location -is	(usually) highest -point
may-be-type	convertible

٣ - يمكن وضع الوصف السابق في شكل يعتمد على القواعد (Rule-based) وذلك يتشغيرها في الشكل : إذا توفر الشرط - تكون النتيجة ، كمايلي:

```
if (? entity)
    has-part spokes and is round
    or (? entity)has -part fancy -hub-cap and is round
then (? entity) may be wheel.
and if (? entity)
    has -attributes wheels owner roof
then
    (? entity) is a auto
    or (? entity) is a mobile- home
    or (? entity) is a .....
    or (? entity)is unknown.
```

٤ - أو وضع وصف هذا الكيان في شكل قائمة للحقائق (Fact Structure) كمايلي:

```
(fact 2 (entity auto)
  (has (wheels owner exterior interior roof engine)
  (other_entries (car truck vehicle))
  (used_for (transportation status _symbole)))
```

٥ - أو وضعها بشكل تركيب قاعدي (Rule Structure) كمايلي:

```
(if ((? entity) has (wheels and owner and roof ....))
(then ((? entity) is a auto)
or ((? entity) is a mobile-home).....)
```

٦ - أو إستخدام قواعد الانتاج (Rule Production Structure) كمايلي:

```
(rule 50 (concerns (car an auto mobile-home))
  (applicable_if (in vehicle frame))
  (antecedent_is
    (if( (? entity) has (wheels and owner and
      roof))
    (production_is
      (then ((? entity) is an auto)
      (alternate_productions
        (then ((? entity) is a mobile-home).....
        (further_information
          (top_down (see transportation status _symbole))
          (bottom_up (see wheels driver owner....)))
          (confidence (use function an auto _confidence)
          (update_rule (apply learning_function
            number 20)))))))
```

(٧-١٥) خواص التمثيل بلغة ليسب

(Characterization of LISP Reperesentations)

كما سبق يتضح الخواص الآتية عند إستخدام لغة ليسب في التمثيل:

١ - غالبا ما يكون التمثيل لبيانات غير رقمية.

٢ - قدرة اللغة على تمثيل بيانات معقدة التركيب والوصف وذلك بإستخدام دالة تقسيم القوائم (car) التى تسمح بتمثيل القوائم البسيطة والقوائم المعقدة مثل الوصف الكامل للسيارة او الدبابة بسهولة ويسر.

٣ - يمكن للتمثيل ان يشمل كيانات مختلفة مثل :
(spoks, wheel, car, owner, tire, roof)

٤ - يمكن للتمثيل ان يشمل العلاقات بين هذه الكيانات المختلفة مثل :
(has, is, has_spokes, may_be).

٥ - يمكن للتمثيل ان يشمل الخواص مثل (is_round , convertible).

٦ - يصبح التمثيل عديم القيمة اذا لم يحتوى على العديد من العلاقات التى تبرز التمييز بين الكيانات والصفات والعلاقات نفسها.

٧ - يجري تمثيل بيانات الصور والكيانات بتركيبات تتميز بانها شبيهة باللغة الانجليزية

٨ - يسمح التمثيل فى شكل القوائم بالوصول الى كفاءة فى طرق البحث والموائمة (Searching, Matching) علاوة على قابلية التحول (Isomorphic) من الرسوم الجرافيكية والبيانية والصور والتراكيب المنطقية.

ومن أهم مميزات التمثيل بلغة ليسب مايلي:

١ - السهولة فى تمثيل القوائم فى ذاكرة الحاسب (Computer memory).

٢ - قابلية التحول (Isomorphic) بين القوائم والتفرع الشجرى (Trees).

٣ - يساعد تمثيل الكيانات فى شكل قوائم على سهولة الحصول على الشبكات الدلالية (Semantic Nets) والتمثيل بإستخدام الاطارات (Frames).

(١٥-٧-١) نماذج ذاكرة الحاسب (Computer Memory Models)

يمكن تقسيم ذاكرة الحاسب الى النماذج الآتية :

١- النموذج القياسى (Standard Model) : الذى يمثل ذاكرة الحاسب الحديث

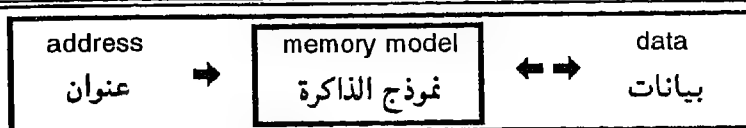
(١٩٩٢) وذلك للقراءة أو الكتابة والمبين فى شكل (٧-١٥) حيث يمكن تمثيل الذاكرة

علي انها مصفوفة ذات بعد واحد (1-D Array) مفهرسة بعناوين (Address) ، وتصبح

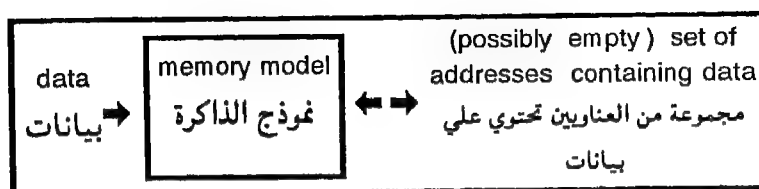
القراءة هى استخراج ما بداخل العنوان من بيانات، والكتابة هى عملية التخزين لهذه

البيانات كما فى شكل (٧-١٥).

٢ - النموذج الترابطى (AM) (Associative Model) والذى يستخدم كثيراً لتسهيل



شكل (٧-١٥) النموذج القياسي (Standard Model) للذاكرة



شكل (٨-١٥) النموذج الترابطي (Associative Model) للذاكرة

عملية البحث والذي يعتبر عكس النموذج الاول من حيث وجود البيانات عند المدخل سوف يرتبط بعناوين فارغة أو عناوين محتوية على بيانات (Content Addressable Memory) (CAM) والمبين في شكل (٨-١٥)

(٢-٧-١٥) تمثيل الرسوم البيانية في الذاكرة

(Computer Representation of Graphs)

لتمثيل الرسوم البيانية في شكل قوائم يتم تخزينها داخل ذاكرة الحاسب يمكن الاستعانة باحدى الطرق الآتية:

١ - المصفوفات الخطية (linear Arrays) : يمكن خزن قائمة في ذاكرة الحاسب بشكل مباشر وذلك بترتيبها لتخزن بأماكن بالذاكرة تأخذ شكل مصفوفة خطية فإذا كان حجم القائمة معروف مقدماً فإن ذلك لا يمثل مشكلة علي الاطلاق، كما هو مبين في شكل (٩-١٥).

من أهم معوقات إستخدام المصفوفات الخطية لتراكيب البيانات مايلي:

١ - يعتبر إضافة أو حذف بيانات أو تغيير حجم التركيب أو نوعيته عملية صعبة.

٢ - لا يعكس التركيب الخطي في الذاكرة أي علاقات بين عناصر القائمة.

٣ - عند إستخدام طرق البحث الثنائية لإستخراج أو إسترجاع عنصر من قائمة مكونة من عدد

(m) من العناصر فإنه يلزم العدد الآتي من المقارنات:

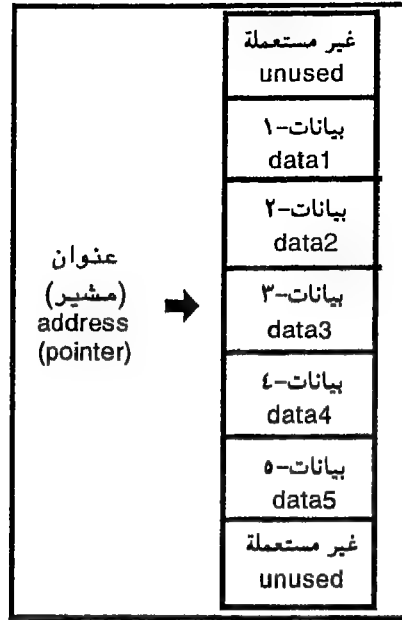
$$n = \log_2(m)$$

٢ - التركيب للقوائم المتصلة (Linked Lists) : من أهم تراكيب البيانات التي لا تسمح

بالمعوقات السابقة هي التراكيب المتصلة والمبينة في شكل (١٠-١٥) حيث يمثل كل عنصر من

التركيب المتصل للقوائم على خلية للبيانات متبوعة بمشير للخلية القادمة والتي تتبع الأولى،

وبذلك يمكن القول بأن الذاكرة في هذه الحالة تصبح ديناميكية من حيث تعيين مكان عنصر



شكل (١٥-٩) تراكيب معتمدة علي المصفوف الخطي البسيط (Simple linear Array)

البيانات، وتصبح الاضافة أو الحذف معتمدة على المشيرات فقط.

من خواص تراكيب البيانات المعتمدة على القوائم المتصلة مايلي:

١ - تبلغ عدد المقارنات في المتوسط لادخال او استخراج عنصر من قائمة متصلة يبلغ عدد عناصرها (m) ما يلي :

$$n \text{ (insert / remove) } = m / 2$$

٢ - يبلغ متوسط عدد المحاولات للإسترجاع العشوائى لعنصر مايلي :

$$n \text{ (retrieval) } = m / 2$$

٣ - يستخدم المشير (END) للإشارة الى انتهاء القائمة.

(١٥-٧-٣) تمثيل العلاقات فى شكل مصفوفات

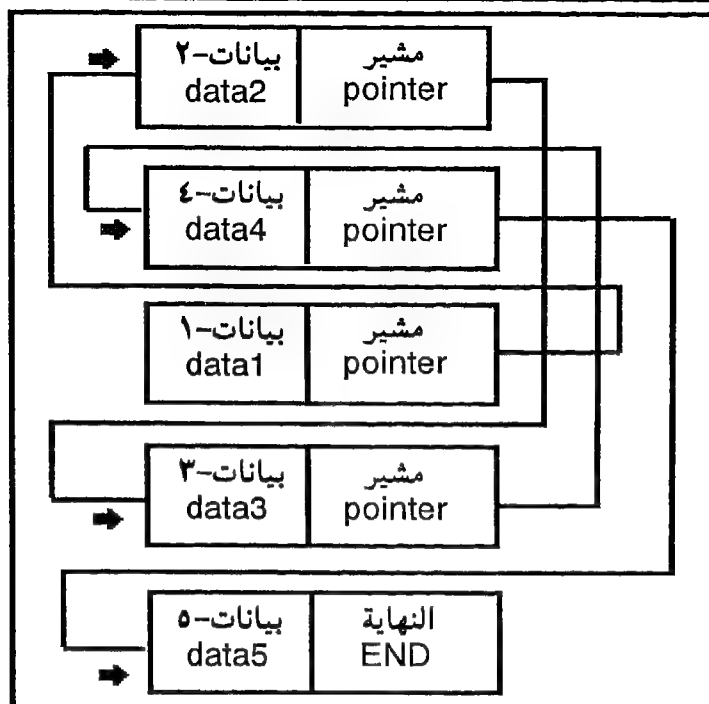
(Matrix Representation of Relations)

إذا فرضنا ان الكيانات والاهداف والفروض والعلاقات التى تشرح موضوع ما كانت معرفة،

وأردنا ان نحصل على تمثيل لهذه التراكيب فى شكل مصفوفة، فإنه يمكن عمل الآتى:

١ - ترقيم كل هدف او شئى والذي يمثل عقدة (node) وبذلك نحصل علي مجموعة من العقد المرسومة التي تحمل مجموعة الارقام [1,2,...,n].

٢ - لتوضيح العلاقة بين أى عقدتين وهو ما يسمى بايجاد الحد بين أى عقدتين وذلك للمجموعة



شكل (١٥-١٠) تراكيب معتمدة على القوائم المتصلة
(Linked - list structure)

المرتبطة بالعلاقة (R) فإنه يمكن تمثيلها بالمصفوفة الآتية :

$$ADJ[i,j] = \begin{cases} 1 & \text{if } (i,j) \text{ in } R \\ 0 & \text{if } (i,j) \text{ not in } R \end{cases}$$

وبذلك نحصل على تركيب للبيانات في بعدين (2D A) أى مصفوفة ثنائية، ويمكن القول بأن هذا التركيب الثنائي البعدين سوف يتم تحويله الى متجه ذو البعد الواحد (1-D) حيث يمكن خزنه بذاكرة الحاسب. وتكمن المعوقات لهذا التمثيل فيما يلي :

١ - تعتبر عملية الاضافة من العمليات المعقدة .

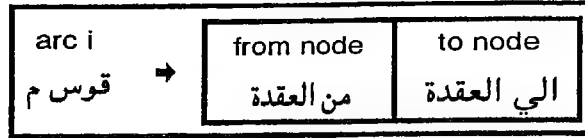
٢ - يعتبر التمثيل غير معنون .

٣ - تعقيد عملية البحث.

٤ - كبر حجم الجزء المحجوز من الذاكرة مع قلة البيانات المخزونة.

وللتغلب على هذه المعوقات يتم الاتى:

١ - اقتراح تمثيل يحفظ العلاقة بين الحدود الفاصلة (الاقواس) والتي تأخذ شكل المصفوفة الاحادية (1-D) من [] الى وذلك طبقاً للقاعدة التالية [] كمايلي:



شكل (١١-١٥) التركيب التمثيلي للأقواس بين العقد

if edge i represent member (x,y) of relation R
then {make the following assignment}
from [i] := x ;
to [i] = : y

وبذلك يتم تكوين تركيب لكل الاقواس في العلاقات والتي يمكن ان تأخذ الشكل المبين في شكل (١١-١٥) والذي يشتمل على تركيب يمثل الاقواس (arc) التي تربط بين العقد، وبذلك فإنه يمكن تمثيل العلاقة الموجهة (R) بخليتين تخصص الخلية الاولى للاتجاه من العقدة (from node). وتمثل الثانية الاتجاه الي العقدة التالية (رقم العقدة التالية) (to node). ويطلق على هذا التركيب " الخلية " (cell) والتي لها الخواص الآتية :

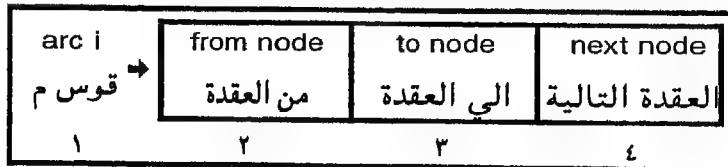
١ - تعتبر الخلية معنونة (Addressed) او مشيرة (Pointed) الى او مفهرسة (Indexed) بالقوس الذي يربط بين عقدتين.

٢ - يمثل جسم الخلية بيانات اسم عقدة المنبع (From) واسم عقدة الوصول (To) كما هو مبين في شكل (١١-١٥). والتي يمكن ان تعدل لتأخذ الشكل المبين في شكل (١٢-١٥)، حيث يتمثل في إضافة جزء ثالث هو رقم العقدة التالية، فإذا اعتبرنا أن هناك أربعة أعمدة من اليسار الي اليمين فإننا نرى ان العمودين الاول والرابع يمثلان الحدود او الاقواس، بينما يمثل الثانى والثالث أرقام العقد.

(٨-١٥) خلية أداة الربط (cons) وبناء القوائم

(The cons Cell and Building Lists)

من الواضح في التمثيل المبين في شكل (١٢-١٥) والذي يتمثل في استخدام القوائم المتصلة (linked lists) لتمثيل المعرفة ان له كثير من الايجابيات التي من أهمها الاضافة او



شكل (١٢-١٥) التركيب التمثيلي المعدل للأقواس بين العقد

car	cdr
خلية الرأس	خلية الذيل

شكل (١٥-١٣) أداة الربط (cons)

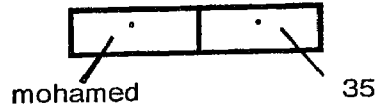
الحذف للبيانات وبذلك يزيد او يقل حجم البيانات طبقا لسير المعالجة.
ومن أهم السليبات لهذا التمثيل هو إعتباره تمثيلا للحواف او الاقواس ولذلك يطلق عليه التمثيل المركزي الحواف (Edge - centric) وفيما يلي نورد تركيب آخر يعتمد علي العقدة والذي يسمى التركيب المركزي العقدة (Node-centric) واستخدام التفرع الشجري الثنائي (Binary Tree) في ذلك. من المعروف ان التفرع الشجري الثنائي لا يحمل اكثر من فرعين. يبين شكل (١٥-١٣) أداة الربط (cons) التي تتركب من خلية للرأس (car) و خلية للذيل (cdr) ويمكن للرأس (car) ان تحتوي علي اي شئ مثل اسم رمزي او قيمة او مشير، بينما يحتوي الذيل (cdr) من دالة الربط (cons) علي مشير لخلية ربط أخرى، وبذلك فإن في هذا التمثيل الممتد يمكن للخلية المثلثة لاداة الربط (cons) ان تحتوي علي مشيرين.

تعتبر خلية الذيل محتوية علي (صفر) (Nil) اذا لم تشر الي شئ.
مثال : إذا أردنا ان نستخدم خلية دالة الربط في تمثيل الحقيقة التالية :
الاسم الرمزي محمد - العمر ٣٥ سنة

نفرض اننا سوف نقوم بإستخدام نظام العنوان المباشرة للذاكرة لتمثل خلية الربط (cons) وبذلك فإننا سوف نقوم بخزن الاسم الرمزي محمد في العنوان المكاني (٤) من الذاكرة وسوف نقوم بخزن العمر (٣٥) في العنوان المكاني (٢) من الذاكرة واننا سوف نستخدم العنوان المكاني الخامس والسادس كخلية للربط لخزن العنوان (٤) والعنوان (٢) وبذلك تظهر الذاكرة كما في شكل (١٥-١٤). ويمكن القول بان هذا التمثيل سوف تصبح بشكل افضل كما هو مبين في شكل (١٥-١٥) حيث يقوم مفسر ليسب (LISP Compiler or Interpreter) بالعناية بالعناوين والاماكن لتأخذ الحية لاداة الربط (cons) الشكل المبين . وبإستخدام التعبير النقطي للتعبير عن هذه الدالة وذلك بوضع نقطة في رأس التمثيل الشجري كما هو مبين في الشكل (١٥-١٦).

1	2	3	4	5	6
	35		mohamed	4	2

شكل (١٥-١٤) إستخدام دالة الربط في التمثيل



شكل (١٥-١٥) الشكل النقطي لاداة الربط (cons)



شكل (١٥-١٦) التمثيل الشجري لاداة الربط (cons)

(٩-١٥) التعبير النقطي او الثنائيات المنقطة

(LISP "Dot " or Dotted Pair Notation).

يعتبر التعبير الخطي النقطي من التعبيرات المشتركة بين الانماط المختلفة للغة ليسب. ويجري تمثيل دالة الربط (cons) من بداية الطرف الايسر للقوس ثم مفرد بيانات - مكان خالي - نقطة - مكان خال - مفرد بيانات - الطرف الايمن من القوس، وبذلك يمكن كتابة دالة الربط في الشكل:

(mohamed . 35)

(mohamed . 35)

المقابل للقائمة

ولكي يكون التمثيل تاماً لا بد ان ينتهي بالصفـر وذلك لاعطاء الحاسب تنبيهاً بان القائمة قد انتهت ويتم ذلك من خلال وضع الذيل (cdr) للخلية (cons) صفراً (nil).

مثال : لتمثيل القائمة (a (b c) d) في شكل دالة الربط (cons) والمبين في التمثيل الشجري شكل (١٥-١٧) حيث يتضح ان النهاية للقائمة الكبرى (d, nil) والقائمة الصغرى (c, nil). ويمكن لدالة الربط (cons) ان تحتوى علي دالة ربط أخرى (cons) وتمثيلها شجرياً، فلوضع اسم العائلة في المثال السابق فإن ذلك يمكن تمثيله في الشكل (١٥-١٨) والذي يتم وضعه في التمثيل النقطي الآتي:

((Mohamed . Aly) . 3.5)

((((Mohamed Aly) 35)

وفي شكل قائمة كالآتي:

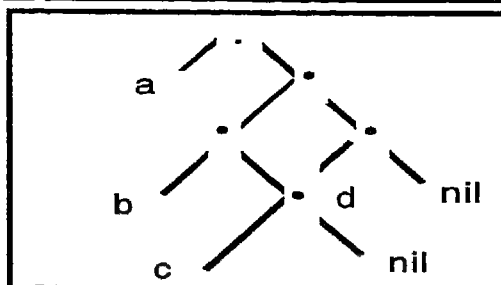
وإذا اضفنا ان صفة مثل (male) للقائمة لتصبح كما هو مبين بالتمثيل الشجري المبين في

شكل (١٥-١٩) والذي يقابل التشكيل النقطي :

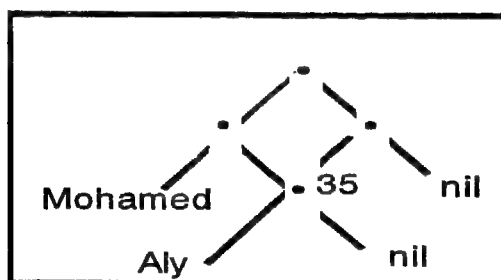
((((Mohamed . Aly) . 3.5) . male)

والذي يمكن تعديله في التمثيل الشجري المبين في شكل (١٥-٢٠) والذي يقابل مايلي:

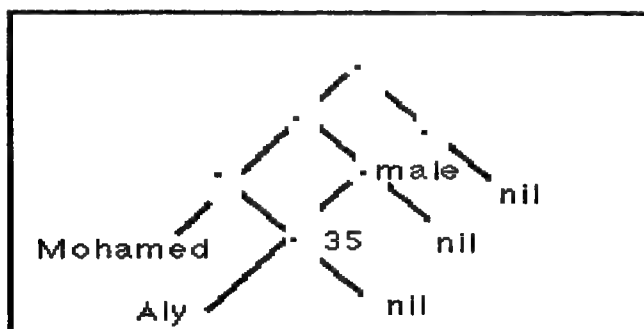
((((Mohamed . Aly) . (3.5 . male))



شكل (١٧-١٥) التمثيل الشجري للقائمة (a b c d)

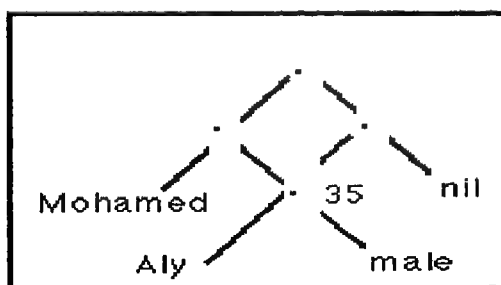


شكل (١٨-١٥) التمثيل الشجري للقائمة ((Mohamed . Aly) 3.5)



شكل (١٩-١٥) التمثيل الشجري للقائمة

((Mohamed . Aly) . 3.5) . male)



شكل (٢٠-١٥) التمثيل الشجري للقائمة

((Mohamed . Aly) (3.5 . male))

الجزء الرابع

الشبكات العصبية الاصطناعية والحساب العصبي

Artificial Neural Networks
and Neural Computing

الفصل السادس عشر

تطور

الشبكات العصبية الاصطناعية

Artificial Neural Networks
Development

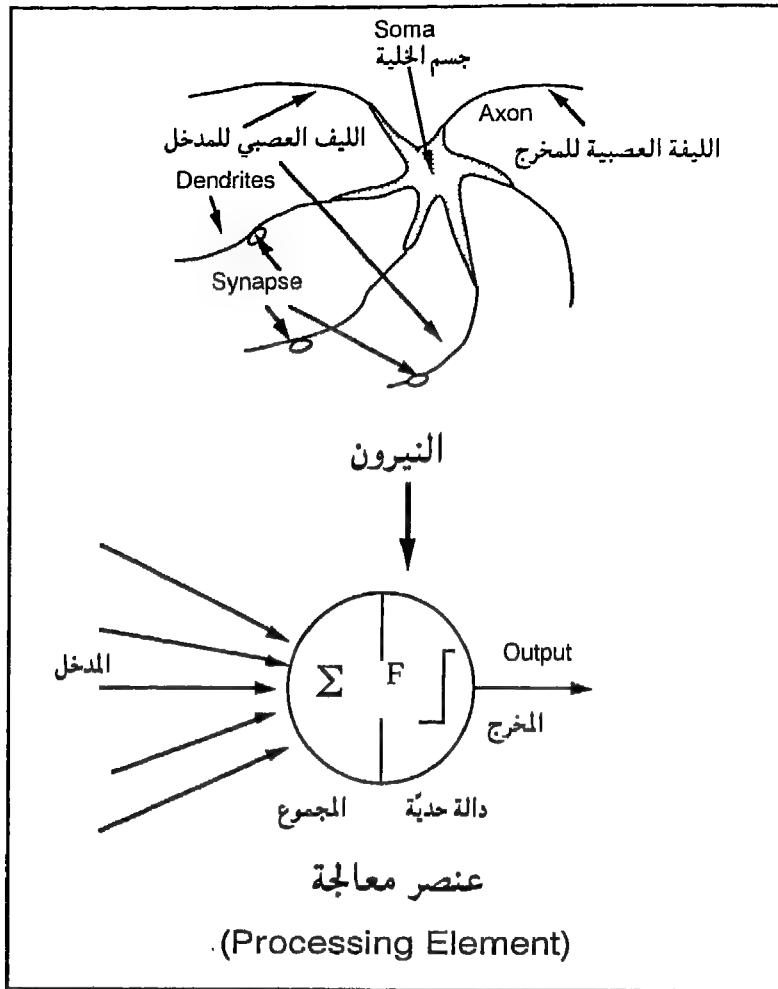
(١٦-١) الشبكات العصبية الاصطناعية (Artificial Neural Networks)

بالنظر فى شكل (١-٣) من الباب الاول والذى يبين العلاقة بين الذكاء البشرى والشبكات العصبية الاصطناعية، نرى أن دراسة طرق عمل المخ والخلايا العصبية قد أدى الى ظهور وتطور الشبكات العصبية الاصطناعية. من المعروف أن الجهاز البشرى للمعالجة يقوم أساسا على الخلايا العصبية فى المخ والى تعتمد على الخلية العصبية النيرون (Neuron) كوحدة بنائية لها.

يبين شكل (١٦-١) تمثيل مبسط للخلية العصبية (النيرون)، حيث يتكون التركيب من جسم الخلية (Soma) والليف العصبى لمدخل الخلية (Dendrites) والذى يحمل الاشارات من المخارج لخلايا عصبية اخرى الى مداخل هذه الخلية عن طريق مشابك (Synapses) للتوصيل، ثم الليفة العصبية الوحيدة للمخرج (Axon) والى تحمل النبضة الخارجة من الخلية الى خلايا اخرى متعددة، ومن المعروف أن جسم الخلية وماحولها يحتوى على الايونات الاتية : الصوديوم والكالسيوم والبوتاسيوم والكلوريد، ومن المعروف كذلك أن ايونات البوتاسيوم تتركز داخل جسم الخلية، بينما تتركز ايونات الصوديوم خارجها، وعندما تحدث إثارة كهربائية (Stimulation) للغشاء الخارجى لجسم الخلية والذى غالبا مايكون انخفاضا في الجهد، فان الغشاء يقوم تبعا لذلك بالسماح لايونات الصوديوم والكالسيوم بالمرور من خلاله الى جسم الخلية حيث يؤدى ذلك الى تغير فى حالتها الداخلية لتعطى نبضة من خلال الليفة العصبية للمخرج.

يمكن عقد التماثل (Analogy) الكهربى الذى يوضح انه يمكن اعتبار الليف العصبى للمدخل والمخرج على أنها تمثل موصلات معزولة ذات معاوقة كهربية (Impedance) مختلفة القيم والى تحمل تبعا لذلك نبضات الى جسم الخلية (Schwartz 1985, Arbib 1964) وتحتوى الشبكة العصبية فى الانسان على بلايين الخلايا العصبية (النيورونات) المتصلة والمتفرعة مع بعضها البعض، حيث تقوم الألياف العصبية للمداخل بإستقبال وحمل النبضات التى ترسلها المخارج، حيث يقوم على تنظيم نقلها المشابك.

لقد تم تطوير نموذج عنصر حسابى (Processing Element) مكافئ لهذا النيرون والموضح فى شكل (١٦-١) والذى يمثل عنصر المعالجة الذى يقوم بعمل محاكاة للنيرون الطبيعى من تجميع للإشارات الموزنة عند المدخل ثم مقارنة المجموع بقيمة حدية داخلية (Threshold)، حيث يعطى عنصر المعالجة نبضة فى المخرج اذا زاد المجموع عن القيمة الحدية ولا يعطى أية نبضات اذا كانت أقل منها، ولقد تم استخدام عناصر المعالجة هذه فى بناء الشبكات العصبية الاصطناعية.



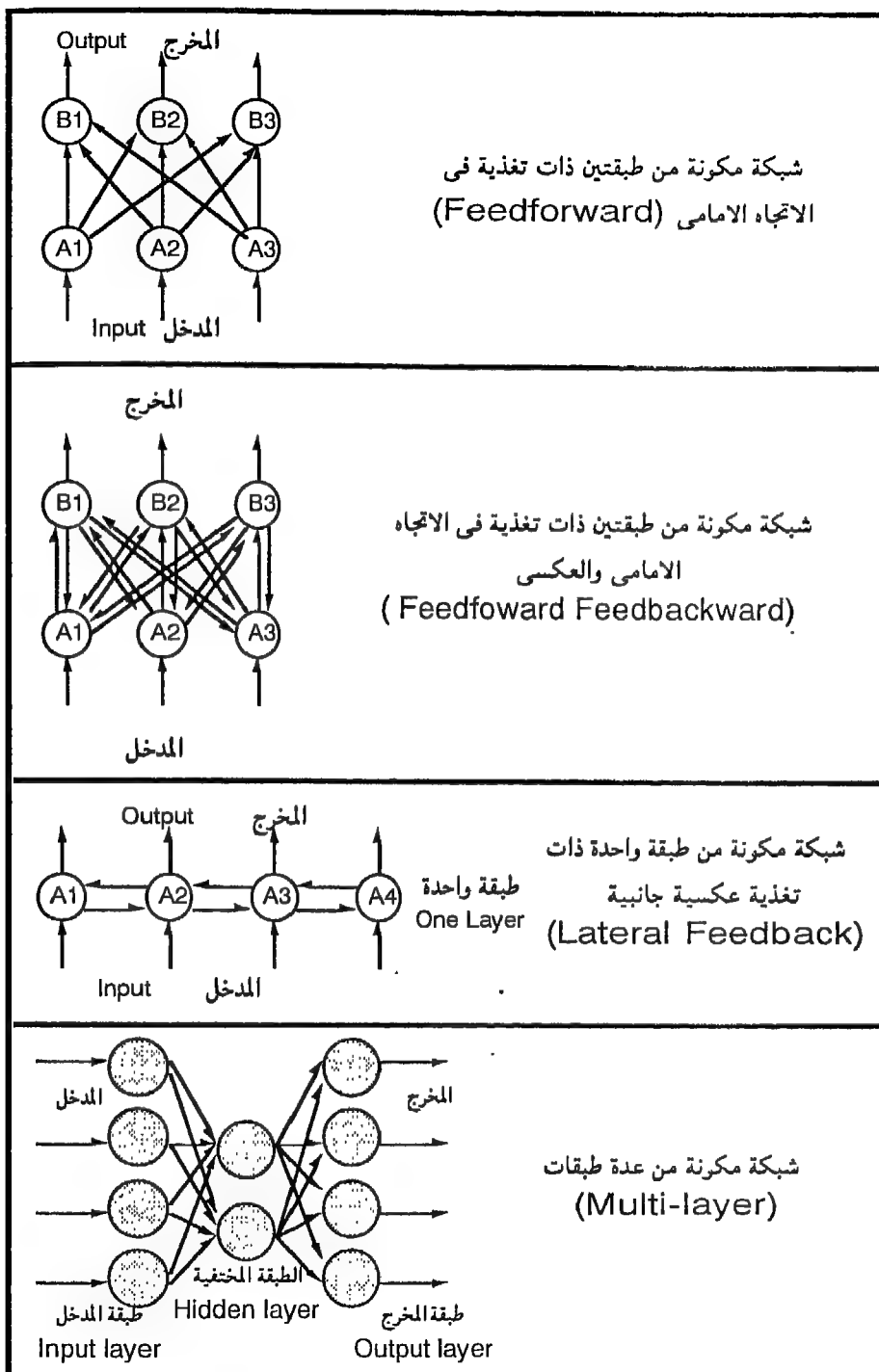
شكل (١٦-١) التمثيل المبسط للخلية العصبية والنموذج الحاسبي المكافئ

فما هي الشبكات العصبية الاصطناعية ؟

يجيب على هذا السؤال كل من أريبب وجوسبرج ونلسون :

(Arbib 1964, Grossberg 1982-1986, Hecht-Nielson 1988) كما يلي:

" ان الشبكات العصبية الاصطناعية (Artificial Neural Networks) هي تركيبات للمعالجة المتوازية الموزعة (Parallel Distributed Processing Structure) تعتمد أساسا على عنصر المعالجة (P.E.) القادر على العمل كذاكرة محلية (Local Memory) مع اجراء عمليات المعالجة المختلفة، والذي له مخرج واحد يتفرع الى كثير من التفرعات (Fans Out) التي تحمل نفس الاشارة الخارجة منه مع بقاء المعالجة محلية، أى انها تعتمد على القيم المدخلة



شكل (١٦-٢) بعض أمثلة للمعماريات المختلفة للشبكات

وكذلك القيم المخزونة بالذاكرة المحلية (Local Memory) لهذه العناصر الحسابية " .

يمكن وضع تصور فلسفى آخر لتعريف الشبكات العصبية الاصطناعية وذلك بوصفها على انها رسوم بيانية موجهة (Directed Graph) ذات حواف موزنة (Weighted Edges) قادرة على خزن الاشكال والبصمات (Patterns) وذلك بتعديل قيم الاوزان للحواف، وبذلك يمكن لهذه الشبكات ان تتعرف على هذه البصمات او الاشكال مرة ثانية اذا كانت القيم المدخلة غير كاملة (Incomplete) او غير مُعرفة (Unknown).

تشارك معظم الشبكات العصبية الاصطناعية فيما يلى:

١- التمثيل الموزع (Distributed Representation)

٢- المعالجة المحلية (Local Processing)

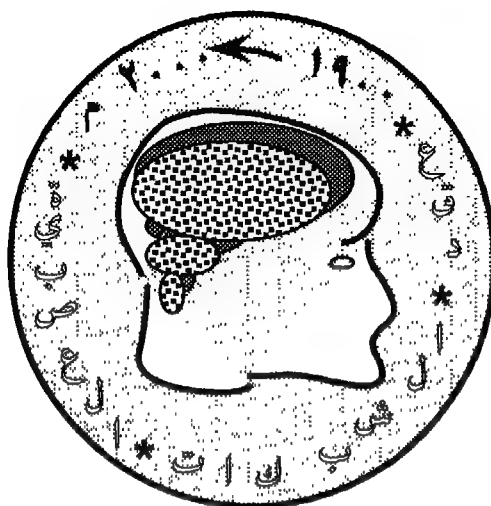
٣- المعالجة اللاخطية (Nonlinear Processing) .

يبين شكل (١٦-٢) بعض أمثلة لمعماريات مختلفة من الشبكات مثل : شبكة مكونة من طبقتين ذات تغذية فى الاتجاه الامامى (Feedforward) وشبكة مكونة من طبقتين ذات تغذية فى الاتجاه الامامى والعكسى (Feedbackward-Feedforward) وشبكة مكونة من طبقة واحدة ذات تغذية عكسية جانبية (Lateral Feedback) ومعمارية الشبكة المتعددة الطبقات (Multi-layer).

(١٦-٢) تطور الشبكات العصبية الاصطناعية

(Artificial Neural Networks Development)

يمكن القول أن بداية تاريخ التفكير فى الشبكات العصبية الاصطناعية قد بدأ فى القرن الماضى حيث قام فرويد (Freud) بالتعرض الفلسفى للفكرة العامة لها، ويعتبر أول تطبيق عملى للشبكات البدائية فى عام ١٩١٣ حيث نفذ روسل (Rusel) جهازا هيدوليكيًا معتمداً على الفكرة العامة لهذه الشبكات، ويمكن اعتبار فترة الاربعينات (١٩٤٠-١٩٥٠) هى البداية الحقيقية لتطور هذه الشبكات حيث ساهم الكثير من العلماء والمهندسين فى تطويرها، وتعتبر التسعينات من هذا القرن هى القفزة الحقيقية فى التطوير، حيث أعلن البيت الابيض الامريكى ان العقد الحالى (١٩٠٠-٢٠٠٠م) هو عقد المخ والشبكات العصبية والحساب العصبى، وذلك نظرا للزيادة الكبيرة فى عدد المهتمين بها والذين حضروا مؤتمرات الجمعية الامريكية الفيزيائية (American Physical Society) اعوام ١٩٨٥، ١٩٨٦، ١٩٨٧، والذى ارتفع عددهم من ٦٠ شخصا الى اكثر من ٢٠٠ شخص فى الاعوام التالية (١٩٨٨-١٩٩٠) بعد ذلك.



العقد الحالي (١٩٠٠-٢٠٠ م) من هذا القرن هو عقد الشبكات العصبية الاصطناعية

وفيما يلي التسلسل الزمني لأهم الانجازات في تطوير الشبكات العصبية الاصطناعية.

ماك كلوش و بتس (MCCULLOCH & PITTS) ١٩٤٣:

وضع أول نموذج رياضى للنيرون او العصبون والذي أدخل فكرة الدالة الحدية (Threshold Function) والتي اعتبرت اساسا للنظم التي جاءت بعد ذلك مثل نموذج هوبفيلد (١٩٨٢) والذاكرة الترابطية ثنائية الاتجاه (Associative Bidirectional Memory) لكوسكو (Kosko) (١٩٨٨)، ومن اهم السبلبيات لهذا النموذج هو عدم القدرة على التعلم.

هيب (HEEB) ١٩٤٩ :

تم تطوير نظام رياضى للتعلم للشبكات العصبية والذي يحمل اسم التعليم الهيببى (Hebbian learning) وتقول هذه النظرية مايلى: " تؤثر حالة النشاط السابقة (Postsynaptic activity) للمشابك (Synapses) عند اعادة تنشيطها فتزداد كفاءة التوصيل او الشدة (Strength) لهذه المشابك اذا استقبلت نبضة جديدة متلاحقة لنبضة اخرى سابقة ". فاذا فرضنا ان جهد المشبك موجب وذلك نتيجة لنبضة سابقة ثم جاءت نبضة اخرى بفرق زمنى صغير فان الموصلية (Conductance) لهذا المشبك تزداد وعلى العكس اذا كان جهد المشبك سالباً ثم جاءت نبضة اخرى بفرق زمنى صغير فان الموصلية لهذا المشبك تقل. وتعتبر نظرية التعلم هذه من الاساسيات التى استخدمت لتطوير الشبكات بعد ذلك.

مارفن مينسكى (MARVIN MINSKY) ١٩٥١:

قام منسكى عام (١٩٥١) بالعمل على تصميم آلة قابلة للتعليم تحتوى على ٤٠ نيرون متخذاً نموذج ماك كولوش اساساً لبنائها حيث يتم ضبط الموصلية للمشابك طبقاً لنجاح الآلة فى تنفيذ عمل معين وذلك بتطبيق نظرية التعلم الهيببى. وفى عام (١٩٦٠) قام بدراسة النموذج المقترح من فرانك روزنبلات والمسمى عنصر الادراك العصبى (Perceptrons) وايجاد حدود التطبيق له واثبت ان المستقبل العصبى المكون من طبقتين لا يقدر على ايجاد الحلول للمشكلات التى توصف بحلول منفصلة بشكل خطى (Linearly separable solutions) وقام باقتراح الشبكات العصبية المتعددة الطبقات (Multilayer).

أوتلى (UTTLEY) ١٩٥٦:

قام باقتراح نظرية لعمل آلة تحتوى على معالجات مزودة بفواصلات خطية (Linear sep-arators) يعتمد عملها على ضبط المتغيرات المدخلة باستخدام نظرية شانون (Shannon) والتى استخدمت فى التمثيل للنظم العصبية الحقيقية.

روزنبلات (ROSENBLATT) ١٩٥٧:

قام بتطوير نموذج ماك كولوش للنيرون وذلك باضافة نظرية التعلم واطلاق اسم عنصر الادراك العصبى (Perceptron) عليه ودراسة النماذج ذات الطبقتين والثلاث طبقات المكونة من عناصر الادراك واقترح نظرية عناصر الادراك المجمعة (Perceptron convergence theorem) والتى تؤدى الى ضبط الاوزان بين المدخل والمخرج طبقاً لقيم الخطأ بين قيم المخرج المطلوب والمحسوب بالشبكة ولكنه لم يتوصل الى طريقة رياضية واضحة لتعليم الطبقة المختلفة.

ودرو (WIDROW) ١٩٥٩:

قام ودرو باقتراح شبكات عصبية شبيهة بعناصر الادراك اطلق عليها اسم العناصر الخطية المتكيفة ذاتياً (Adaptive linear element or Adaline) والتى تقوم بضبط وتعديل الاوزان بين طبقة المدخل والمخرج طبقاً للفرق بين المطلوب (Desired) والمحسوب (Computed) وفى عام (١٩٦٠) تم اثبات ان الفرق او الخطأ بين المخرج المطلوب والمحسوب يصل الى قيمة صغرى عند شروط معينة. ولقد تم تطبيق هذه الشبكات فى كثير من التطبيقات المختلفة مثل معالجة الاشارات المتكيفة ذاتياً (Adaptive signal processing) ونظم التحكم ونظم الهوائيات المتكيفة ذاتياً (Adaptive antenna). وفى عام (١٩٨٨) تم تطوير خوارزميات للتعلم الاختيارى المتكيف (Selective bootstrap adaptation).

شتينبوش (STEINBUCH) ١٩٦١

قام شتينبوش بتطوير نظرية التشفير (Encoding) فى الشبكات العصبية المتقاطعة القضبان (Crossbar ANS) واستنتاج مصفوفة التعلم (Learning matrix) التى استخدمت فى التعرف على الاشياء الكبيرة التشويه مثل الكتابة اليدوية واغراض التحكم فى الصناعة.

جروسبرج (GROSSBERG) ١٩٦٤

تعتبر ابحاث جروسبرج من اهم الاعمدة التى ساهمت فى تطوير نظم الشبكات العصبية الاصطناعية المتكيفة (Adaptive systems) حيث قام بإنشاء مركز للنظم الانضباطية المتكيفة بجامعة بوسطن. ولقد قام بدراسة العمليات الفسيولوجية والبيولوجية بالمخ وقام بالربط بين المخ والعقل واشتقاق نظرية لذلك.. ومحاولة تصميم شبكات عصبية قادرة على التنظيم الذاتى (Self-organizing) والاتزان الذاتى (Self-stabilizing) والقدرة الذاتية على تغير المقياس وذلك للعمل فى الزمن الحقيقى ولقد أدت ابحاثه الى تصميمات تركيبية أقل ما يمكن من التفرع للشبكات العصبية تأخذ شكل النجمة المتفرعة الى الداخل (Instar) او النجمة المتفرعة الى الخارج (Outstar) وذلك للتعلم والتعرف على البصمات واعادة صياغتها ومعالجتها (١٩٦٨-١٩٨٢) وتم اشتقاق طريقة التعلم التنافسى المتعاون (Competitive-cooperative) وكذلك نظرية الرنين الانضباطى المتكيف (ART) والرنين الانضباطى المتكيف الثنائى والتماثل. ومن اعوام (١٩٨٣) الى (١٩٨٩) قام مركز الابحاث تحت قيادته بتطوير الكثير من الموضوعات مثل الخواص الديناميكية للشبكات والامكانية العامة للتحويل من شبكة الى اخرى وكذلك دراسة الرؤية والتحدث ووضع الخواص العامة لنظرية الرنين الانضباطى الثنائى (ART1) والتماثل (ART2). ومن أهم سمات التطوير فى هذا المعهد هو احتواء التنفيذ العملى للشبكات على التماثل للمعالجة الإنسانية.

أمارى (AMARI) ١٩٦٢

قام بدراسة الخواص الديناميكية للشبكات المتصلة عشوائيا والشبكات غير التماثلة الاتصال.

اندرسن (ANDERSON) ١٩٦٨

منذ (١٩٦٨) الى (١٩٨٦) قام اندرسون بدراسة الشبكات العصبية التى تحتوي علي نماذج للذاكرة المترابطة (Associated memory) واستخدام الدوال الحدية الخطية والمتجهات فى ذلك. وقام باقتراح جمع البصمات المتعددة على نفس الذاكرة لتعطى الذاكرة ذات الترابط الخطى

(LAM) وكذلك قام باقتراح الشبكات العصبية التى تسمى (BSB) حالة المخ فى صندوق والتى تعتمد على استخدام التغذية العكسية الجانبية لتصحيح الاخطاء للتعلم واستبدال الدالة الحدية بدالة منتظمة الزيادة والتى تقوم بخزن المعلومات فى الزوايا لمكعبات متعددة المقياس.

فيوكيوشيما (FUKUSHIMA) ١٩٦٩

بدأ فيوكيوشيما العمل فى ابحاثه عن الشبكات العصبية فى (١٩٦٠) ببناء الشبكات العصبية المتعددة الطبقات والخاصة بالاستخدام فى الرؤية ثم طورها بعد ذلك باستحداث الشبكة العصبية المسماة المتعرف الادراكى (Cognitron) عام (١٩٧٩) ثم طورها الى المتعرف الادراكى العصبى (Neucognitron) عام (١٩٨٧) وهى شبكة عصبية متعددة الطبقات والتى يبلغ عددها تسعة طبقات حيث تقوم كل طبقة باستخلاص السمات من الطبقة التى قبلها واستخدام الطبقات المختفية اللازمة والتى نجحت فى التعرف على الارقام المكتوبة باليد والمشوهة والمدارة والمختلفة الاحجام.

كلوف (KLOPF) ١٩٦٩

قام بوضع نظرية ونموذج للتعلم اطلق عليها اسم الطريقة التفاضلية للتعلم الهيبباني (١٩٨٥) (Differential Hebbian learning).

كوهنن (KOHONEN) ١٩٧١

قام بالتركيز على دراسة الذاكرة الترابطية الخطية (Linear associative memory) والتى تحتاج الى متجهات خطية حرة (Independent linear vectors) حيث تقوم بخزن انسب المتجهات التى ما تكون غالباً غير خطية والتى اطلق عليها الذاكرة الترابطية الخطية المثلى (Optimal Linear associative memory). ويستخدم هذا النوع من الذاكرة فى الخزن المتوسط، كما انه يستخدم كمرشح الذى يقوم بمقارنة المتجهات المدخلة بالمتجه المخزون. ولقد قام كوهنن بتطوير طرق للتعلم التسابقي او التنافسى (Competitiv learning) والذى يسمى التعلم الكمى للمتجهات (Learning vector quantization) والذى يحدد اوتوماتيكيا المتجه المناسب من عدد كبير من المتجهات، والتى تقوم كذلك بتنظيم نفسها. ولقد تم استخدام هذه الشبكات (LVQ) فى التعرف على الحديث والصور.

ليون كوبر (مجموعة نستور) (NESTOR ASSOCIATES) COOPER ١٩٧٣

اشتغل كوبر منذ السبعينات بدراسة النمذجة للشبكات العصبية ثم قام بإنشاء مجموعة نستور (١٩٨٧) لانتاج شبكات عصبية على المستوى التجارى حيث قامت باستخدام نظرية الطاقة المخفضة لكولوم (Reduced Coulom Energy) لتصميم طريقة نستور للتعلم وتم

استخدام الشبكات المطورة (RCE) فى التعرف على الحروف والتشخيص الصناعى والتعرف على الاهداف والامضاءات.

سيجنووسكى (SEJNOWSKI) ١٩٧٦

بدأ العمل فى ابحاث الشبكات العصبية باستخدام علوم الرياضيات والبيولوجيا منذ عام (١٩٧٦) وذلك بتطوير الشواهد المنطقية لنظرية التعلم، ومن اهم انجازاته اقتراح خوارزميات آلة بولتزمان (Boltzman Machine) وتطوراتها ذات الرتب العالية ثم تطبيق هذه الخوارزميات فى مشاكل الرؤية (١٩٨٦). ولقد قام سيجنووسكى باستخدام نظرية الانتشار الى الخلف (Backpropagation) فى الربط بين نص ومنطوقه والذي لاقى نجاحاً كبيراً وفى عام (١٩٨٨) استخدم نفس النظرية لتصنيف اشارات السونار والتي تم تصنيفها بجودة تفوق الجهد الانسانى المعروف فى ذلك التصنيف (١٩٨٨). واخيراً تم استخدامها فى التنبؤ بتركيب البروتينات والتي تفوقت فى ذلك.

ماكلياند (مجموعة PDP) (PDP group) MCCLELLAND ١٩٧٧

اهتمت هذه المجموعة بدراسة الشبكات العصبية الاصطناعية لفهم طبيعة عمل المخ الانسانى وذلك من خلال نظام التفهم للكلام (Hearsay) والذي يمكن اعتباره نظام نصف متوازى وتحوارى للقراءة. وتم تطوير نظام نصف متوازى لتحليل العمليات العقلية (Mental processes). ويعتبر النموذج التنشيطى التحوالى للتعرف على الكلمات (١٩٨٢) (Interactive activa-tion model) اساس معمارية المعالجة المتوازية الموزعة (PDP) وتم إنشاء مجموعة المعالجة المتوازية الموزعة للابحاث وتم نشر كتاب من جزئين يحتوى الجزء الاول علي نظرية المعالجة المتوازية الموزعة باستخدام الشبكات العصبية الاصطناعية، ويحتوى الجزء الثانى على تطبيقاتها. ومن اهم التطبيقات هو تنفيذ نموذج للحدث.

ستون وبارتو (SUTTON & BARTO) ١٩٧٨

بدأ الابحاث فى (١٩٧٨) فى تطوير اعمال كلوف فى التعلم وتطوير نظرية التعلم الاجبارى (Reinforcement learning) (١٩٨٤) حيث تم توضيح الفرق بينه وبين التعلم باستخدام طريقة تصويب الخطأ (Error - correction) كمايلي:

فى التعلم باستخدام تصويب الخطأ (Error - correction) فان النظام يتذكر ما امر بفعله، اما فى الحالة الثانية (Reinforcement learning) فان النظام يكتشف بطريقة ما الافعال او الشروط التى تؤدى نتائجها الى تحسين النتائج. وتم استحداث طريقة التعلم التى تعرف باسم (Associative reward - penalty).

١٩٨٠. FELDMAN & BALLARD (CONNECTIONIST GROUP) ويلارد فيلد مان

قامت هذه المجموعة بتطوير الكثير من الشبكات العصبية للتطبيقات المختلفة، وخصوصا فى مجال تطبيقات الرؤية بالحاسب، ومعالجة اللغات الطبيعية، والشبكات الدلالية، والاستدلال المنطقى (١٩٨٦).

هيشت- نلسن (HECHT-NIELSEN) ١٩٨٢

يعتبر نلسن رائد أبحاث الحاسب العصبى حيث قاد التصميم لأول حاسب عصبى حديث تحت اسم (TRW Mark III) والذي تم تسويقه فى عام (١٩٨٦) ثم قام ببناء حاسب شخصى تم تسويقه تحت اسم (ANZA) عام (١٩٨٧) ثم قام بتطوير اعمال كل من جروسبرج وكولموجورف، وفى عام (١٩٨٨) قام بتطوير خوارزميات التجمع للإنتشار المرتد Convergence of Backpropagation Algorithm ويعتبر من أوائل الاشخاص الذين ساهموا فى ادخال الشبكات العصبية الى التطبيقات اليومية.

هوفيلد وتانك (HOPFIELD & TANK) ١٩٨٢

قام هوفيلد بتحديد النقطة المستقرة (Stable point) للشبكات العصبية ذات القضبان المتقاطعة (Crossbar) عام ١٩٨٢ ولقد تم بناء النموذج على تعريف الطاقة المسماة طاقة ليبنوف (Lyapunov) وذلك بالاستعاضة عن القيام بوضع الحل لمجموعة من المعادلات الديناميكية اللاخطية. ويعتمد نموذج هوفيلد على القدرة على بناء معادلة للطاقة التى يمكن ان توصف نشاط شبكة عصبية مكونة من طبقة واحدة فى الزمن المتقطع لجميع المداخل الممكنة وحساب الطاقة المهدرة فان النظام يستقر فى بؤرة محلية متدنية (اقل ما يمكن) (Minimum local) ويبقى ثابتا فيها. ثم قام بتطوير هذه النظرية ليتم تطبيقها فى الزمن المستمر بدلا من الزمن المتقطع، كما تم استحداث معادلة للطاقة التى توصف مشاكل لإختيار الحل الامثل مثل مشكلة البائع المتجول (Salesman problem) ولقد تم تطوير هذه الطريقة للتعرف على البصمات والتحدث المتغير مع الزمن.

ميد (MEAD) ١٩٨٥

يعتبر ميد من أوائل الرواد فى النظم الالكترونية العالية التكامل (VLSI) وذلك بمحاولة تطبيق النظم العصبية للانسان فى المجال الالكترونى، حيث قام بعمل نموذج لشبكية العين (Retina) من مادة السيلكون عام (١٩٨٨) حيث احتوت النبطية على (٢٣٠٤) خلية حساسة ضوئيا والذي دفعة ذلك الى محاولة تقليد الجهاز العصبى الكامل عام (١٩٩٠).

كوسكو (KOSKO) ١٩٨٥

قام كوسكو بتطوير ثلاثة محاور رئيسية فى مجال الشبكات العصبية وهى:

- ١ - الذاكرة الترابطية الثنائية الاتجاه (BAMs) (Bidirectional Associative Memories) ذات القدرة على التعلم الغير موجهة والتي يمكنها الاستدعاء أو التذكر (Recall) فى الزمن الحقيقى وامكن تطويرها للوصول الى وضع ديناميكى مستقر للتعلم وتقوم بالاستدعاء فى نفس الوقت .
- ٢ - خرائط فزى للتعرف (FCM) والتي تستخدم المزيج من طريقة التعلم التفاضلى لهيب وطريقة التعلم لكوسكو و كلوف.
- ٣ - تطبيق منطق فزى فى الشبكات العصبية وذلك لاستحداث ما يسمى بذاكرة فزى الترابطية عام (١٩٩٢) (Fuzzy Associative Memory)

هوانج و ليبمان (HUANG & LIPPPMAN) ١٩٨٧

حيث قاما بالمقارنة بين الشبكات العصبية الاصطناعية والمصنفات التقليدية (Conventional classifiers)

فورسيث (FORSYTH) ١٩٩٠

تم وضع ومقارنة الطرق المختلفة للتعلم.

شاركى (SHARKEY) ١٩٩٢

تنظيم التركيبات للشبكات العصبية الاصطناعية ونقل المعارف.

روس ج ماكسويل (Ross J.Maxwell) ١٩٩٣

تم التفرقة بين عينات الانسجة الحية السليمة والمریضة التى تحتوى على أورام، وذلك بتحليل البيانات المأخوذة من جهاز الرنين النووى المغناطيسى وذلك بإدخالها على شبكات عصبية مدربة على تصنيف الأنسجة طبقا لحالتها الصحية .

ارنجتون (ERRINGTON) ١٩٩٣

التعرف والتفرقة الأوتوماتيكية بين الانواع المختلفة للكروموزومات .

تامبوراتزى (TAMBOURATZIS) ١٩٩٣

تطوير استخدام نظرية هارمونى (Harmony Theory) فى تخليق الشبكات العصبية القادرة على اجراء التماثل للمناظر المرئية (Scene Avalysis) .

إمانويل (EMMANUEL) ١٩٩٣

المراقبة المستمرة لنهضات دقات القلب بعد الإجهاد والتعرف على التشخيص للضربات القاتلة..

هينز (HINES) ١٩٩٣

بناء أنف الكترونية معتمدة على الشبكات العصبية الاصطناعية والمجسات المختلفة (Sensors) للتعرف على والتفرقة بين الروائح المختلفة والأبخرة للمواد التى تدخل فى كثير من الصناعات.

هاجن (HAGEN) ١٩٩٣

التعرف على الأخطاء التى يمكن أن تحدث فى الانظمة المختلفة للتحكم الاوتوماتيكي.

ريفنز (REFENS) ١٩٩٣

تستطيع الشبكات العصبية والتي يمكن تغذيتها يوميا بالتغيير فى معدلات اسعار العملات أن تتنبأ على معدلات هذا التغيير فى المستقبل وذلك بناءً على التغييرات السابقة كما يجرى إستخدامها فى التعرف والتنبؤ بالدورات الإقتصادية واتجاه هذه الإقتصاديات على المستوى العالمى.

رينيه بيوالد (RENE BIEWALD) ١٩٩٣

تم تطوير انظمة ملاحية باستخدام هذه الشبكات يمكنها تفادى العقبات أثناء الملاحه فى الاماكن الضيقة، وذلك من خلال انسان آلى تمثل الشبكة العصبية التى تعلمت من قبل العمود الفقرى لهذا النظام .

آخرون ١٩٩٦-١٩٩٤

يعتبر التعرف من أهم التطبيقات للشبكات العصبية الاصطناعية حيث يمكن تدريب وتعليم هذه الشبكات على التعرف على أشكال معينة للشبكة أن تتعرف عليها بعد ذلك أوتوماتيكيا حتى فى وجود شوشرة وعدم وضوح مثال ذلك:

التعرف على البصمات وملامح الوجهة فى المجال الأمنى وتحليل الصور الجوية المأخوذة بالاقمار الصناعية وتصنيف البيانات الزراعية من شكل الاوراق والتعرف على الكتابة اليدوية والامضاءات فى اعمال الشيكات والبنوك والقراءة والترجمة الاوتوماتيكية والتعرف على الصور الناتجة من التفاعلات النووية عالية الطاقة (High Energy Physic) و التعرف على الاشخاص بالصور السينمائية والتليفزيونية المتكررة من خلال تعليم الشبكات على جزء من ملامح الشخص فقط (العين مثلاً).

(General Features of Neuron) السمات العامة للنيرون (٣-١٦)

يعتبر المخ الانسانى من أعقد الموضوعات التى لم يتم كشف أدق أسرارها ودراستها بالتفصيل الى الآن وذلك بالرغم من معرفة الكثير من التركيب والصفات، ويحتوى المخ على:

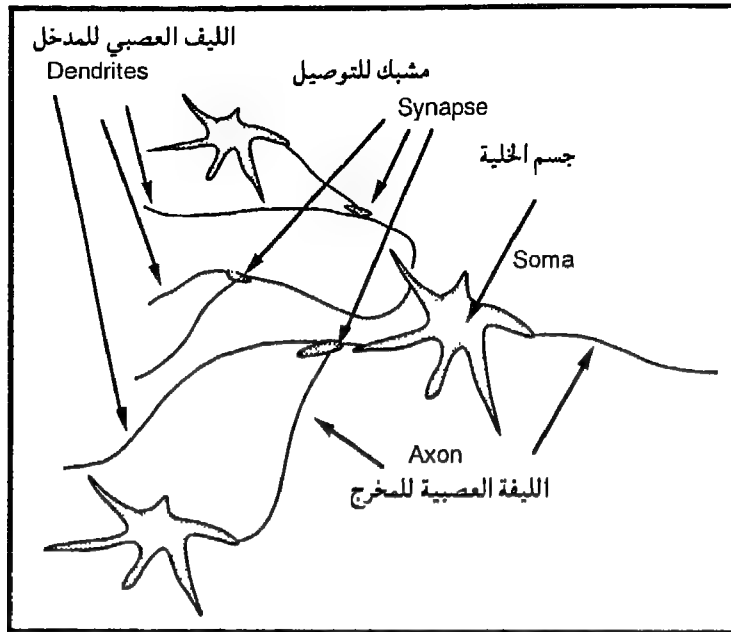
- ١- عشرة بليون (١٠^{١٠}) وحدة خلية عصبية تسمى النيرون (Neuron).
- ٢- يتصل كل نيرون بما يقرب من عشرة آلاف آخرين .
- ٣- يعتبر النيرون الوحدة الاساسية للمخ وحدة معالجة ثنائية منطقية مستقلة (Stand Alone Logical Processing Unite).
- ٤- تتصل كل خلية عصبية بكثير من الالياف العصبية التى تمثل المداخل المتعددة للخلية (Many Inputs) وليفة عصبية واحدة عند المخرج (One Output).
- ٥ - يظل النيرون خاملاً (Inactive) أى لا توجد نبضة عند المخرج اذا كان مجموع النبضات اللحظى عند المدخل لا تصل الى مستوى حدى معين (Thershold) لاثارته ويعطى النيرون نبضة عند المخرج ويصبح نشطاً (Active) اذا زاد المجموع عند المدخل عن المستوى الحدى المعين.

(١-٣-١٦) التركيب المبسط للوحدة العصبية النيرون

(Simple Structure of Neuron)

يبين شكل (٣-١٦) التركيب المبسط للوحدة العصبية النيرون والتى تتكون من الآتى:

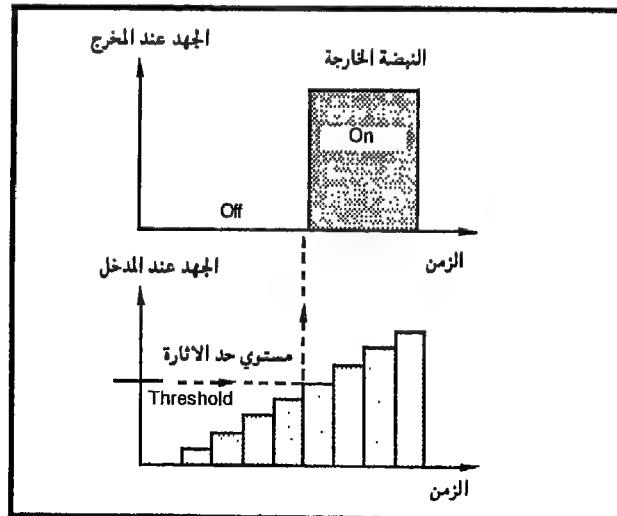
- ١- جسم خلية النيرون والمعروف باسم السوما (Soma).
- ٢- الالياف العصبية المتصلة بجسم النيرون عند المدخل والتى تعرف بالالياف العصبية للمدخل او الغصينات (Dendrites) وهى خاملة من الناحية الكهربائية.
- ٣- يقوم جسم الخلية بعمل عملية كيميائية معقدة يمكن اعتبارها من الناحية الكهربائية المجموع الكلى لقيم النبضات عند المدخل .
- ٤- يتصل بجسم الخلية ليفة عصبية واحدة عند المخرج تعرف بالليفة العصبية للمخرج (Axon) والنشطة كهربائياً والتى تعتبر لخطية التصرف والتى تعطى نبضة كهربائية ذات مدى زمنى يبلغ واحد ملى ثانية تعرف بنبضة جهد العمل (Action Potential) وذلك اذا ارتفع الجهد الداخلى لجسم النيرون عن قيمة الجهد الحدى المعين وذلك نتيجة لعملية الجمع اللحظى للنبضات الآتية من الالياف العصبية المتصلة بالمدخل كما هو مبين فى شكل (١٦-٤).
- ٥- تنتهى الليفة العصبية للمخرج بمشبك للتوصيل (Synapse) والذى يربط بين الليفة العصبية للمخرج مع الليفة العصبية للمدخل لنيرون آخر باستخدام الترابط الكيميائى



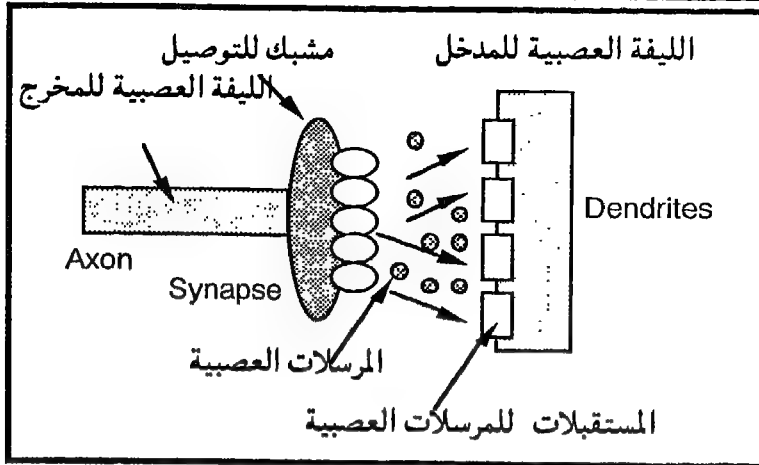
شكل (١٦-٣) التركيب المبسط للخلية العصبية (النرون)

(Chemical Coupling) حيث يقوم المشبك بإطلاق مواد كيميائية تسمى المرسلات العصبية (Neurotransmitter) وذلك عندما يرتفع جهده نتيجة للنبضة العصبية الخارجة من الليفة العصبية للخروج.

٦- ويقوم المشبك بإطلاق المرسلات العصبية التي تنتشر كيميائياً من خلال الفجوة بينه وبين الليفة العصبية لمدخل نرون آخر حيث تقوم بتغيير الجهد لها وتنشيطها ليصبح هذا الجهد



شكل (١٦-٤) الجهود عند المخرج كدالة في الجهود عند المدخل



شكل (١٦-٥) تمثيل مبسط للفجوة بين مشبك التوصيل حيث تنطلق منه
المرسلات الى المستقبلات على الليفة العصبية لمدخل نيرون آخر

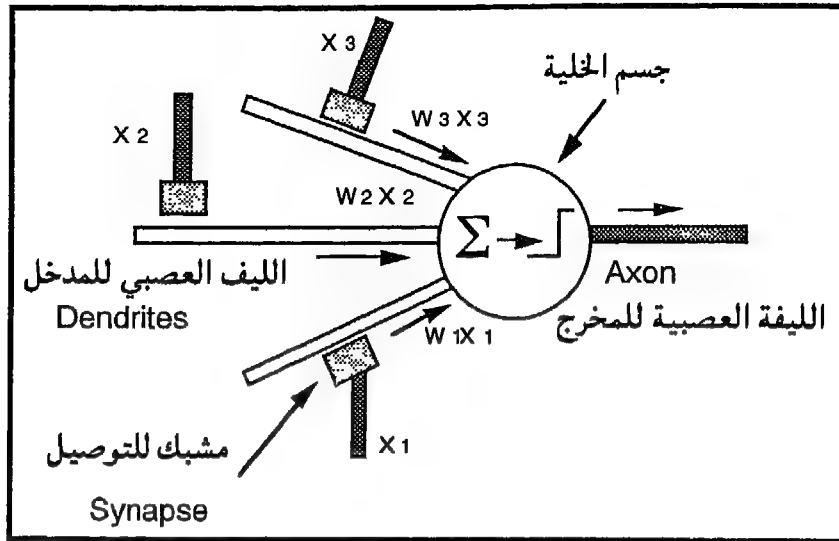
احد المداخل للنيرون الاخير.

٧- وبين شكل (١٦-٤) الجهد عند المدخل كدالة في الزمن ومن الواضح ان تجمع النبضات
لا يؤثر على جهد المخرج الا اذا اصبحت قيمته اكبر من مستوى حد الاثارة فتنتقل النبضة
ويرتفع جهد المخرج وبذلك يمكن القول بأن الخلية العصبية تتبع النظام الثنائي حيث تتغير
حالتها بين حالة الصفر (Off state) والواحد (On state).

(١٦-٣-٢) ميكانيكية التعلم في النظم البيولوجية

(Learning Mechanism in Biological Systems)

يبين شكل (١٦-٥) تكبير لفجوة الاتصال بين المشبك (Synapse) والليفة العصبية
(Dendrite) لمدخل خلية اخرى والذي يوضح ان المشبك يقوم باطلاق المرسلات العصبية
(Neurotransmitters) الى المستقبلات المنتشرة (Receptors) على الليفة العصبية الاخرى
والتي تعمل كبوابات (Gates) وبذلك يتحدد الارتباط بعدد البوابات المستقبلية النشطة ويمكن
القول بان قيمة الارتباط المؤثر (Effective Coupling) لكل مشبك تختلف من فجوة الى اخرى
وان عملية ضبط القيم المختلفة التي تعطى الاتصال المطلوب هي بمثابة تحديد الاوزان
رياضيا (Weight Adjustment). وبذلك يمكن اعتبار النيرون كعنصر معالجة مستقل
(Processing Element) وان هذه المعالجات المستقلة تعمل على التوازي وان عملية التعلم
ماهي الا ضبط لقيم الاوزان عند المداخل لهذه المعالجات والذي يتم على عدة دورات تعرف بدورات
التعلم الى ان يتم الضبط النهائي المطلوب.



شكل (١٦-٦) النمذجة الرياضية لطريقة عمل الخلية العصبية

(١٦-٣-٤) نموذج "ماك كلوش وبتس" الرياضي

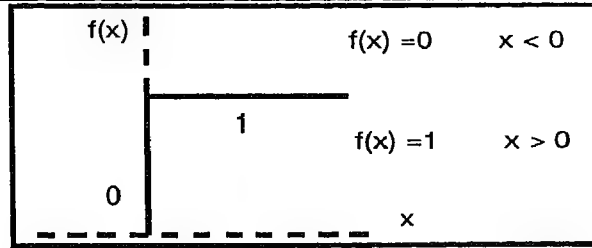
(McCulloch & Pitts Mathematical Model)

لوضع نموذج رياضي لطريقة عمل الخلية العصبية لابد من اتخاذ الاعتبارات الآتية:

- ١ - ان يكون قيمة المخرج من النورون اما الصفر (Off) او الواحد (On).
- ٢ - ان يتكون الدخل من عدد من المداخل بحيث يعتمد المخرج على القيم لهذه المداخل وتكون القيمة للمخرج (واحد) اذا زاد المجموع الموزن (Weighted Sum) لهذه المداخل عن قيمة حدية (Threshold Value) لجهد الاثارة ويظل المخرج (صفرًا) اذا لم يصل المجموع الموزن الى القيمة الحدية المناسبة، وبين شكل (١٦-٦) النموذج المثل للنورون والذي يتكون من:
 - ١- المدخل (Input): نفرض أن قيم مخارج النورونات السابقة للنورون الحالي هي القيم x_1, x_2, \dots, x_n حيث تقوم المشابك بنقل نسبة من هذه القيم الى الليف العصبى للمدخل من خلال الفجوات بحيث تختلف هذه النسب من فجوة الى اخرى اعتمادا على طبيعة كل فجوة، ويمثل ذلك رياضياً بفرض انها اوزان (Weights) والتي تاخذ الشكل w_1, \dots, w_n وبذلك يكون الجزء المنقول الي المدخل فى كل فرع هو $w_i x_i$.

- ٢- جسم الخلية (عنصر المعالجة) (Processing Element): يقوم جسم الخلية بعمل الآتى:
 - (أ) عملية جمع للمداخل الموزنة (Weighted Sum) لتأخذ الشكل :

$$\text{Total input} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \dots = \sum_{i=1}^{i=n} w_i x_i$$



شكل (١٦-٧) دالة هيفيسيد العتبية الشكل

Step Function (Heaviside Function)

(ب) مقارنة هذا المجموع بقيمة حدية لدالة فى شكل عتبية (Step Function)

والمعروفة بدالة (Heaviside) والمبينة فى شكل (١٦-٧)، فإذا كان المجموع

للمداخل الموزنة اكبر من او تساوى الدالة العتبية فان الخرج سوف يكون (١) وإذا

كان المجموع اقل من قيمة الدالة العتبية فان ناتج الخرج يصبح (صفرًا).

٣ - المخرج (Output): يكون الخرج للنموذج الرياضى ام الواحد او الصفر ويمكن ان يستخدم

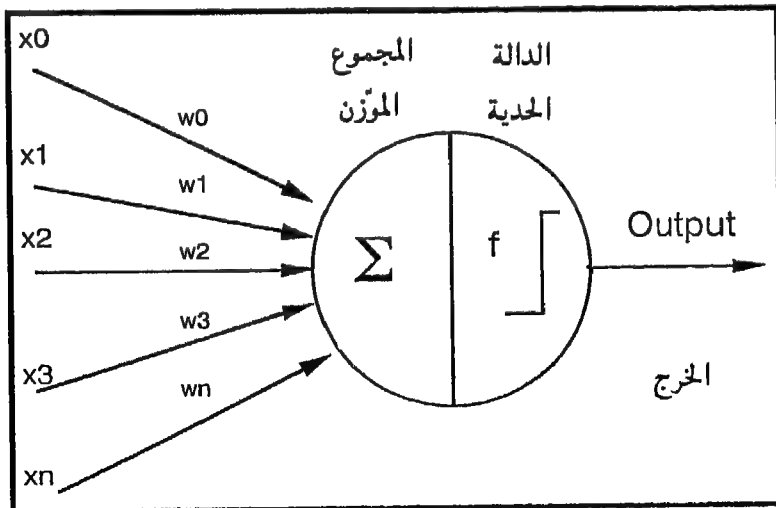
كأحد فروع الادخال لنبيرون آخر، ويمكن كتابة قيمة الخرج (y) فى الشكل الآتى.

$$y = f \left(\sum_{i=1}^n w_i x_i \right)$$

(١٦-٣-٥) النموذج ذو القيمة المنحازة (Biased Model)

ولوضع قيمة انحيازية (θ) لنموذج النبيرون بحيث تمثل القيمة ($w_0 x_0$) شكل (١٦-٨)

والتي تجعل النبيرون دائما نشطاً (الخرج يساوى واحدا فى حالة عدم وجود قيم للمداخل الموزنة)



شكل (١٦-٨) نموذج "ماك كلوش ويتس" الرياضى للنبيرون

وبذلك فان هذه القيمة لابد ان تطرح من المجموع الموزن للمدخل ويقارن الباقي فاذا كان موجبا فان قيمة المخرج تكون واحداً واذا كان الباقي سالباً فان المخرج يكون صفراً وتصبح النتيجة كمايلي:

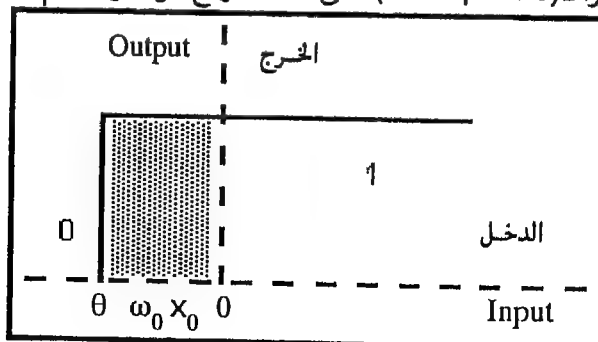
$$y = f \left(\sum_{i=1}^n w_i x_i - \theta \right)$$

حيث يمكن تغيير المجموع ليبدأ من (i=0) كمايلي:

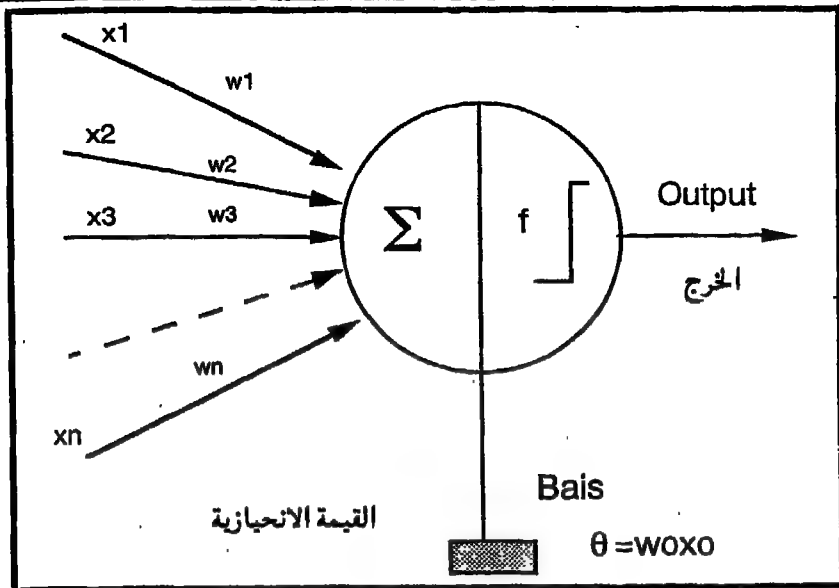
$$y = f \left(\sum_{i=0}^n w_i x_i \right)$$

يبين شكل (١٦-١٠) النموذج الرياضى ذو القيمة الانحيازية ،و يمكن القول بان هذه النماذج تتسم بالخصائص الآتية:

- ١ - يمثل النموذج وحدة معالجة بسيطة (Simple Processing Unit).
- ٢ - باستخدام المجموع الموزن للمداخل ومقارنة هذا المجموع بقيمة حدية يمكن الحصول على المخرج فى صورة الواحد اذا كان المجموع اكبر من القيمة الحدية ،والصفر اذا كان المجموع أقل منها.
- ٣- فى حالة استخدام قيمة انحيازية (θ) والتي تجعل النموذج نشطا اى ان قيمة المخرج تساوى الواحد فانه يجرى طرح هذه القيمة (θ) من المجموع الموزن فاذا كان الناتج اكبر من الصفر او موجبا فان المخرج يعطى واحداً واذا كان الناتج اقل من القيمة الانحيازية او سالباً فان المخرج يكون صفراً .
- ٤ - يعتبر هذا النموذج والذي تم اتخاذه اساسا لبناء الشبكات العصبية الاصطناعية نموذجاً حسابيا فقط وليس نسخة مما يحدث فى المخ الانسانى.
- ٥ - تعتبر اكبر المميزات لهذا النموذج الحساى هو سهولة تنفيذه على الحاسبات، ولقد تم اطلاق اسم عنصر الادراك (Perceptrons) على هذا النموذج من قبل العالم فرانك روزنبلات.



شكل (١٦-٩) العلاقة بين المدخل والمخرج في وجود القيمة الانحيازية



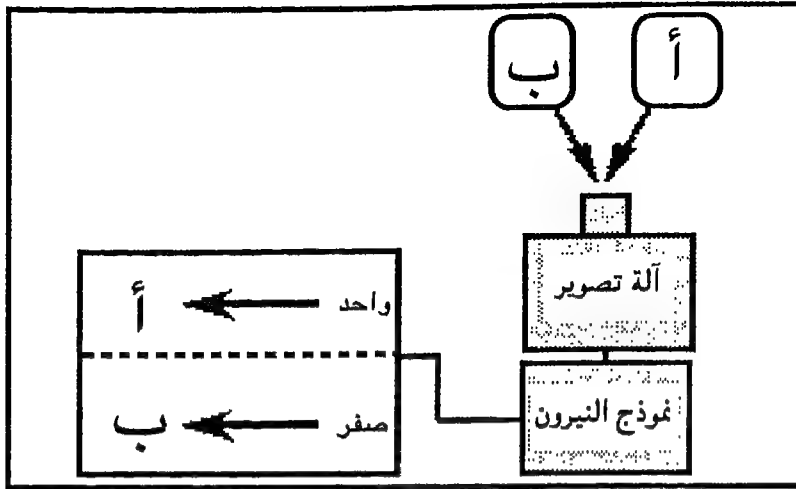
شكل (١٦-١٠) النموذج الحسابي لعنصر الادراك (Perceptron)

(١٦-٤) ميكانيكية التعلم فى النيرونات البسيطة

(Learning Mechanism in Simple neurons)

تعتبر عملية التعلم للشبكات العصبية هى الاساس فى تحويل هذه الشبكات لأداء عمليات التعرف المختلفة والتى تتم بإستخدام اسلوب التعليم او التدريب (Training) للشبكة والتى تقاثل عملية التعليم والتدريب التى تجرى للاطفال لكى يتمكنوا من التعرف على الأشياء. فإذا فرضنا ان هناك شبكة عصبية وهذه الشبكة العصبية متصلة بآلة تصوير، وسوف يتم تعريض مجموعتين من اللوح المكتوب عليها حرف من الحروف، وان هذه الحروف عبارة عن مجموعتان تمثل المجموعة الاولى الحرف (أ) وتمثل المجموعة الثانية الحرف (ب) ويراد تدريب الشبكة لكى تتعرف على هذه المجموعات وذلك بان تقوم الشبكة باعطاء خرج يساوى واحد اذا وضع الحرف (أ) امام آلة التصوير وتعطى صفراً اذا كانت الصورة للحرف (ب) وذلك كما هو موضح فى شكل (١٦-١١).

وبدراسة النموذج الحسابي للنيرون نجد انه من المناسب ان نجعل النيرون يتعلم من اخطائه وذلك باعطاءه فى البداية اوزان عشوائية عند المدخل حيث يقوم النيرون بإجراء عملية الجمع الموزن للمداخل ثم مقارنة المجموع بالقيمة الحدية، فإذا كان المجموع اكبر من القيمة الحدية فإن الخرج سوف يكون الواحد الصحيح ويصبح التعرف على الحرف (أ) ممكننا وإذا كان المجموع اقل من القيمة الحدية فإن الخرج يكون صفراً ولا يتم التعرف.



شكل (١٦-١١) - تمثيل لكيفية تدريب الشبكة العصبية

من المعروف احصائيا ان الفرصة سوف تكون ٥٠٪ : ٥٠٪ في عملية التعرف الاولى فإذا فرضنا ان المجموع الموزن اقل من القيمة الحدية وبذلك لا يعطي الخرج الواحد ويكون المطلوب من تعليم النموذج مرة ثانية وذلك بمحاولة زيادة الاوزان لكي يصبح المجموع الموزن اكبر من القيمة الحدية ويكون الخرج واحدا. وتكرر العملية عدة مرات اخرى في اتجاه زيادة المجموع الموزن الى ان يتم الحصول على الواحد في المخرج.

وعلى العكس عند محاولة التعرف على الحرف (ب) والتي تتطلب ان يكون المجموع الموزن عند المدخل اقل من القيمة الحدية وبذلك نحصل على الصفر عند المخرج والذي يمثل التعرف على الحرف (ب) ، فإذا كان المجموع اكبر من القيمة الحدية فانه يلزم التدريب مرة ثانية في اتجاه تقليل الاوزان ويتم التدريب عدة مرات في هذا الاتجاه الى ان نحصل على الصفر عند المخرج والذي يعتبر التعرف الصحيح على الحرف (ب).

وهذا يعنى اننا اذا اردنا ان نعلم احدى الشبكات العصبية التعرف فإننا نتبع الآتى:

١ - زيادة الأوزان فى المداخل النشطة (Active inputs) والتي تتطلب ان يكون الخرج نشطاً ايضاً. (الخرج يساوى ١) ويجرى تنفيذ ذلك بإضافة قيم المداخل الى الاوزان.

٢ - تقليل الاوزان فى المداخل الغير نشطة (Inactive inputs) والتي تتطلب ان يكون الخرج غير نشطاً، ايضاً (صفر فى المخرج) ويتم ذلك بطرح قيم المداخل من الأوزان.

وتعتبر مراحل التعلم للشبكة هو تنفيذ الخطوة (١) فقط حيث ان الخطوة الثانية لا تؤثر على النتيجة، ويعتبر دونالد هيب اول من وضع نظرية التعلم وذلك بتطوير نظام رياضى للتعلم للشبكات العصبية والذي يسمى التعليم الهيببىانى (Hebbian learning) وتقول هذه النظرية

مايلي:

"تؤثر حالة النشاط السابقة (Postsynaptic activity) للمشابه (Synapses) عند اعادة تنشيطها فتزداد كفاءة التوصيل او الشدة (Strength) لهذه المشابه اذا استقبلت نبضة جديدة ملاحقة لنبضة اخرى سابقة".

فاذا فرضنا ان جهد المشبك موجب وذلك نتيجة لنبضة سابقة ثم جاءت نبضة اخرى بفرق زمني صغير فان الموصلية (Conductance) لهذا المشبك تزداد، وعلى العكس اذا كان جهد المشبك سالباً ثم جاءت نبضة اخرى بفرق زمني صغير فان الموصلية لهذا المشبك تقل، وتعتبر نظرية التعلم هذه من الاساسيات التي استخدمت لتطوير الشبكات بعد ذلك. ويمكن تلخيص عملية التعليم للشبكة كمايلي:

١ - الإختيار العشوائى (Random Selection) للاوزان (Weights) والقيم الحدية (Thershold).

٢ - ضع قيم للمداخل (Present inputs).

٣ - احسب الخرج الحقيقى (Actual) وذلك بطرح القيمة الحدية (Thershold) من المجموع الموزن للمداخل (Weighted sum).

٤ - يتم تغيير الاوزان مرة اخرى فى اتجاه التعرف (Correct Decisions) (اى الحصول على الواحد) وأضعاف الانحياز الاخر (Incorrect Decisions) اى العمل على تقليل الخطأ (Error Reduction).

٥ - وضع قيم اخرى للمداخل وتكرار الخطوات السابقة.

(١٦-٤-١) خوارزم التعليم لعنصر الادراك (Perceptron Learning Algorithm)

توضح الخطوات التالية خوارزم التعليم الذى يمكن تشفيرها باى لغة من لغات البرمجة المناسبة للحصول على برنامج للتعليم :

أ - الخطوة الاولى:

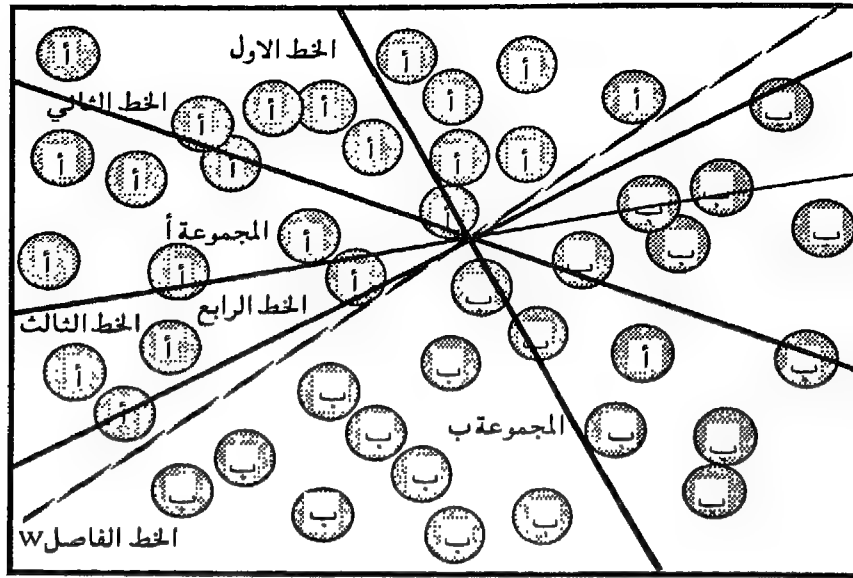
١ - إبدأ عملية التحديد للاوزان والقيم الحدية (Initialise weights and threshold) (شكل (١٦-١٠). كمايلي:

نفرض ان عدد المداخل (n) ولتكن $(x_1, x_2, x_3, x_i, \dots, x_n)$

٢ - يتم اختيار قيمة $(x_0) = 1$ دائماً، وقيمة الوزن $\theta = - (w_0)$

حيث θ = القيمة الانحيازية (Bias)

٣ - يتم الاختيار والتحديد العشوائى لقيم الاوزان عند المداخل عند زمن معين (t) كمايلي:



(١٦-١٢) - تمثيل عملية التعلم في شكل متجه للمجموع المؤزن

$$w_1(t), w_2(t), w_3(t), \dots, w_i(t), \dots, w_n(t)$$

٤- يتم الاختيار والتحديد العشوائى للقيمة الحدية.

ب - الخطوة الثانية : حدد قيم المداخل $(x_1(t), x_2(t), x_3(t), \dots, x_i(t), \dots, x_n(t))$

وكذلك قيمة الخرج المطلوب او المرغوب $d(t) = (\text{Desired})$

ج - الخطوة الثالثة : احسب الخرج الحقيقى كمايلى:

$$y(t) = f \left(\sum_{i=0}^{i=n} w_i(t) x_i(t) \right)$$

د - الخطوة الرابعة :

اجراء الضبط للاوزان (Adapt weights) وذلك كمايلى:

$$w_i(t+1) = w_i(t) \quad \text{١- الحالة المنضبطة :}$$

٢- اذا كان الخرج صفرا ويجب ان يكون واحدا وذلك للتعرف على مجموعة الحروف (أ)

فيجب الوصول الى الشرط الآتى:

$$w_i(t+1) = w_i(t) + x_i(t)$$

٣ - اذا كان الخرج واحدا ويجب ان يكون صفرا وذلك للتعرف على مجموعة الحروف (ب)

فيجب الوصول الى الشرط الآتى:

$$w_i(t+1) = w_i(t) - x_i(t)$$

ولقد تم تطوير هذا الخوارزم بعد ذلك كمايلى:

١ - اجراء التعديل الأول فى عملية ضبط الاوزان وذلك فى الخطوة الرابعة المعدلة أولاً كمايلي:

*د - الخطوة الرابعة المعدلة أولاً:

اجراء الضبط المعدل للاوزان (Adapt weights-modified version) وذلك كمايلي:

$$١ - \text{الحالة المنضبطة } w_i(t+1) = w_i(t)$$

٢ - اذا كان الخرج صفراً ويجب ان يكون واحداً وذلك للتعرف على مجموعة الحروف (أ) فيجب الوصول الى الشرط الآتى:

$$w_i(t+1) = w_i(t) + \eta x_i(t)$$

٣ - اذا كان الخرج واحداً ويجب ان يكون صفراً وذلك للتعرف على مجموعة الحروف (ب) فيجب الوصول الى الشرط الآتى:

$$w_i(t+1) = w_i(t) - \eta x_i(t)$$

حيث يكون الحد (η) موجبا وينحصر بين الصفر والواحد الصحيح والذي يجرى استخدامه للتحكم فى معدل الانضباط.

٢ - ولقد قام (Widrow & Hoff) بتعديل الخوارزم السابق وذلك بهدف ان يكون الانضباط فى تعديل الاوزان بقدر كبير اذا كان الفرق بين الخرج المطلوب والحقيقى كبيراً ويكون صغيراً اذا كان الفرق صغيراً وذلك بإقتراح دالة دللتا للخطأ واقتراح قاعدة التعلم (Widrow - Hoff delta rule) كمايلي:

يعرف الفرق بين المجموع الموزن (Weighted sum) والخرج المطلوب (Desired output) على انه دالة الخطأ كمايلي:

$$\Delta = d(t) - y(t)$$

وبذلك تكون $\Delta = 1$ واحداً صحيحاً اذا كان الخرج المطلوب واحداً وكان المجموع الموزن يساوى الصفر، وتصبح القيمة $\Delta = -1$ اذا كان الخرج المطلوب صفراً والمجموع الموزن يساوى الوحدة، وتصبح $\Delta = 0$ الصفر فى حالة التساوى او التعرف الكامل ويصبح اجراء التعديل فى عملية ضبط الاوزان وذلك فى الخطوة الرابعة المعدلة ثانياً كمايلي.

**د - الخطوة الرابعة المعدلة ثانياً:

اجراء الضبط المعدل للاوزان (Adapt weights Widrow - Hoff delta rule) وذلك كمايلي:

١- الحالة المنضبطة

$$\Delta = d(t) - y(t) = 0$$

$$w_i(t+1) = w_i(t) + \eta \Delta x_i(t)$$

وفى حالة التعرف على الحروف (أ)

$$d(t)=+1$$

وفى حالة التعرف على الحروف (ب)

$$d(t)=0$$

ولقد اطلق (Widrow) على هذا النوع من الخوارزم النيرون الخطى الانضباطى (Adaptive linear neurons) ADALINE.

(١٦-٤-٢) التمثيل المتجهى لميكانيكية التعليم

(Vectorial Analogy for Learning Mechanism)

يمكن استخدام التمثيل بالمتجهات فى تفهم ميكانيكية التعليم لعنصر الادراك البسيط وذلك كمايلى : نفرض انه يمكن تمثيل المدخلات للنيرون فى شكل متجه:

$$X = (x_0, x_1, x_2, \dots, x_n)$$

وانه يمكن كتابة الاوزان فى شكل متجه اخر:

$$W = (w_0, w_1, w_2, w_3, \dots, w_n)$$

وبذلك يمكن الحصول على المجموع الموزن فى شكل حاصل ضرب النقطى لمتجهين (Dot Product) كمايلى :

$$\text{Dot Product} = \sum_{i=0}^n W_i X_i = W \cdot X$$

ومما سبق يتضح ان خوارزميات التعلم تؤكد ان عملية ضبط الاوزان تجري في اتجاه تقليل الخطأ في كل مرة عن المرة السابقة، ويمكن تمثيل عملية التعلم هذه كما فى شكل (١٦-١٠) والذي يمثل مجموعة الحروف (أ) ومجموعة الحروف (ب) وذلك باستخدام الابعاد الثنائية فى الرسم حيث يكون المطلوب هو الخط الذى يفصل بين البصمتين .

وللحصول على هذا الخط فانه يجري خطوات التنفيذ لخوارزم التعلم والمبينة سابقاً حيث يمثل الخط الاول فى المحاولة الاولى الخطوة الاولى من خطوات الخوارزم حيث يمثل الخط متجه المجموع الموزن والذي تم اختياره عشوائياً فى البداية ثم تجرى المحاولة الثانية حيث يتم تقليل الخطأ ثم المحاولة الثالثة والرابعة وفى كل مرة يكون الخطأ اقل من الحالة السابقة الى ان نصل الى المحاولة الخامسة والتي تمثل متجه المجموع الموزن الذى يفصل بين بصمة المجموعة الاولى وبصمة المجموعة الثانية. ومعنى ذلك ان النموذج قد تعلم ان يفرق بين هذه البصمات اذا تم خزن المجموع الموزن الاخير.

(١٦-٤-٣) حدود التطبيق لنموذج عنصر الادراك المفرد

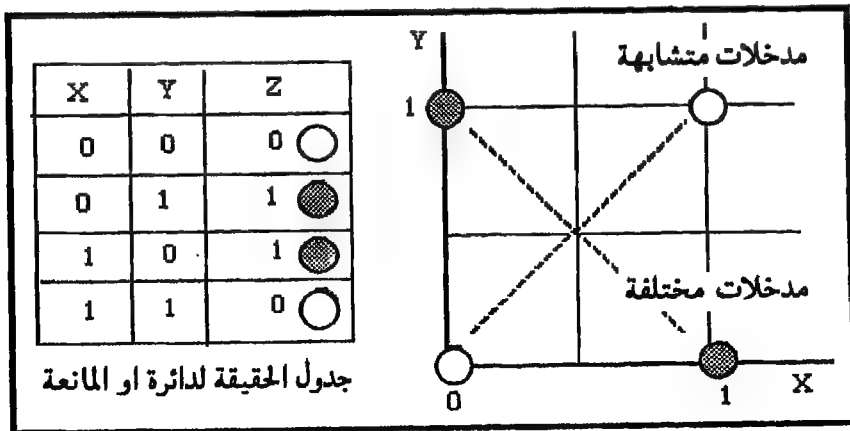
(Limitations of Perceptron Applications)

من الواضح ان النموذج الحسابي لعنصر الادراك المبين سابقا يمكنه الفصل بين مجموعتين منفصلتين فعلا وذلك بوجود حل يفصل بينهما في شكل خط مستقيم ويفشل النموذج في وضع الحل اذا كانت المجموعتين غير منفصلتين بخط مستقيم وذلك واضح في المثال التالي :

نفرض حالة الدائرة المنطقية (أو- المانعة) (XOR) والتي تشتمل على مدخلين (X), (Y) والمخرج (Z) والتي تعطى واحدا عند المخرج ($Z=1$) في حالة ان يكون المدخلين مختلفين (0 or 1) ، وتعطي صفرا ($Z=0$) اذا كان المدخلين متشابهين كما هو واضح في الجدول في شكل (١٦-١٣).

وعند تمثيل هذه الدائرة باستخدام الاحداثيات الثنائية (X,Y) ورسم قيم المخرج (Z) على نفس الرسم فإننا نرى ان هناك مجموعتين الاولى الدائرة الفارغة والتي تمثل ($Z=0$) والتي تنشأ من مدخلات متشابهة والثانية الدائرة المصمتة والتي تمثل ($Z=1$) من مدخلات مختلفة ومن الواضح انه لا يمكن فصل المجموعتين بخط واحد وذلك بسبب وجود خطين متعامدين حيث يربط كل خط بين الدائرتين المتشابهتين .

ويطلق على مثل هذا النوع البصمات التي لا يمكن فصلها بخط مستقيم (Linearly in-separable) . وبذلك فإن نموذج النيرون ذو الطبقة الواحدة لا يمكن أن يعطي الحل للمشاكل التي لا يمكن فصلها بخط مستقيم واحد.



(١٦-١٣) حدود التطبيق لعنصر الادراك المفرد

الفصل السابع عشر

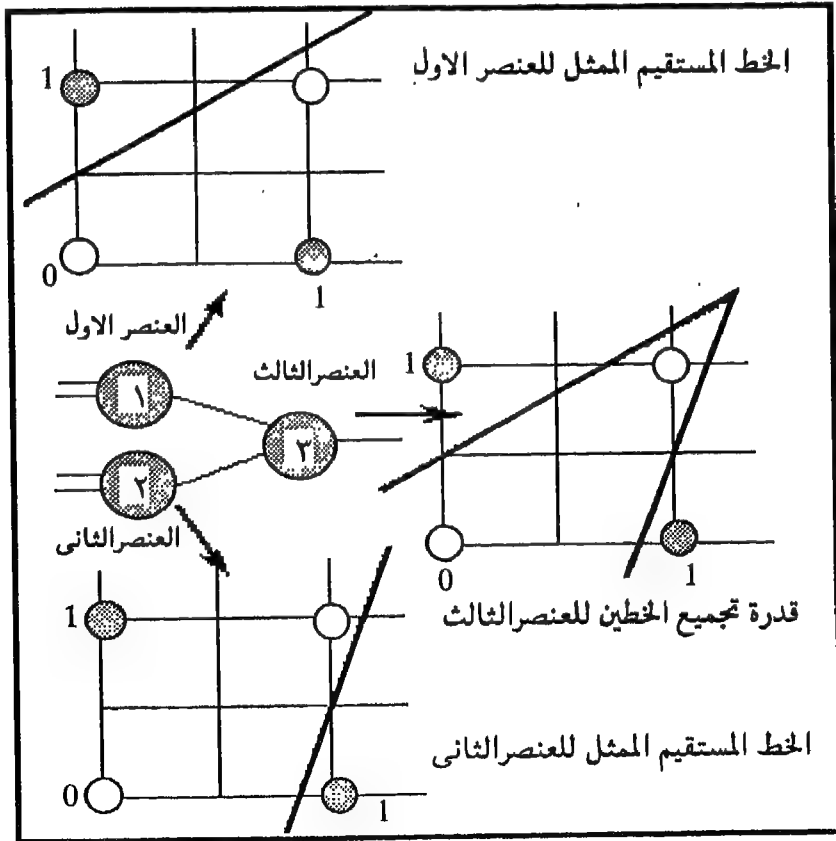
الشبكات العصبية
المتعددة الطبقات

**Multilayer
Neural Networks**

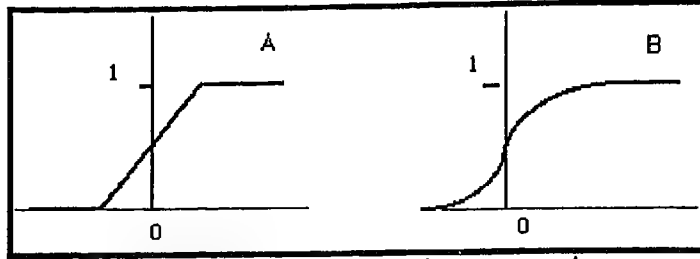
(١٧-١) الشبكات العصبية المتعددة الطبقات

(Multilayer Neural Network)

من الواضح كما رأينا فى الفصل السابق أن النموذج للشبكة العصبية ذات الطبقة الواحدة لا يكون قادرا على حل مشكلة الدائرة (أو) المانعة (XOR) وذلك نظرا لانها غير قابلة للفصل بخط مستقيم واحد، ويمكن استخدام نموذج النيرون ذو الطبقة الواحدة كوحدة بناء لشبكة عصبية مكونة من طبقتين. تحتوى الطبقة الاولى على العنصر (١) و (٢) وتحتوى الطبقة الثانية على العنصر (٣) حيث يقوم العنصر (١) بفصل الخرج ($Z=1$) بالخط المستقيم الاول والذي يكافئ القيم المدخلة (0,1) عن البصمة الكلية، وكذلك يقوم العنصر الثانى (٢) بفصل الخرج ($Z=1$) والذي يكافئ القيم المدخلة (1,0) بالخط المستقيم الثانى عن باقى البصمة وبضم العنصر الثالث الى العنصرين الاخرين يمكن الحصول على الفصل المبين فى شكل (١٧-١) حيث يقوم العنصر الثالث بتجميع الخططين من العنصر الاول والثانى.



(١٧-١) شبكة من طبقتين تحتوى على ثلاثة عناصر وقادرة على وضع الحل لدائرة او المانعة



(١٧-٢) - الدوال الحدية: (A) دالة حدية خطية ذات ميل ثابت ،
(B) دالة حدية مقطعية (Sigmoidal) ذات ميل متغير

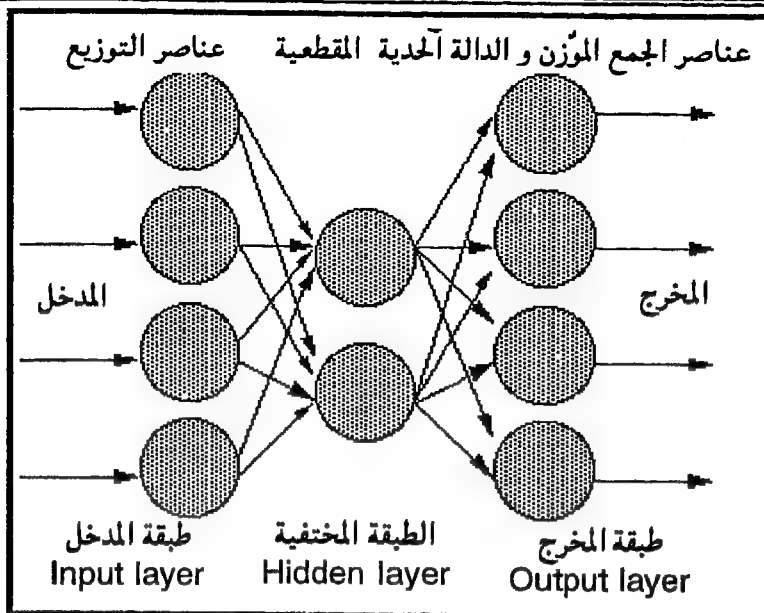
ويبدو للوهلة الاولى ان الشبكة الثنائية الطبقات قادرة على الفصل التام بين البصمات ولكن هناك سلبيات أهمها ان العنصر الثالث في الطبقة الثانية يكون غير قادر علي التمييز بين أى من المداخل النشطة عند المدخل نظرا لانعزالها وبذلك لا يمكن الاجراء الصحيح لضبط الاوزان كما ان الدالة العتبية (Step Function) المثلة للقيمة الحدية في شكلها الجامد لا تساعد على التفرقة و المعرفة الحقيقية لحالة المداخل وكيفية ضبط الاوزان وللتغلب على هذه المشكلة كان لابد من استخدام دالة حدية ذات طبيعة لا خطية (Non-linear) تتمثل في وجود ميل او ميل يتغير ببطئ حيث يساعد هذا الميل على التفرقة في قيم الاوزان الواجب تعديلها.

وبين شكل (١٧-٢) دالة حدية ذات ميل ثابت ودالة مقطعية (Sigmoidal) متغيرة الميل (شكل الحرف S) في اللغة الانجليزية) ويعتبر هذا التطوير هو الاساس في الشبكات العصبية المتعددة الطبقات.

(١٧-١-١) النموذج متعدد الطبقات (Multilayer Model)

يتكون النموذج العام للشبكات المتعددة الطبقات والمبين في شكل (١٧-٣ أ) من ثلاث طبقات كمايلي:

- ١ - طبقة المدخل (Input layer) والتي تعمل عناصرها الحسابية (PE) على توزيع القيم المدخلة ولا تقوم بإجراء عملية الجمع الموزن (Weighted Sum) او وضع القيم الحدية للاثارة.
- ٢ - الطبقة المخفية (Hidden layer) وهى الطبقة التى تقع بين طبقة المدخل وطبقة المخرج وتقوم عناصرها الحسابية بإجراء عملية الجمع الموزن (Weighted Sum) وتطبيق القيم الحدية للاثارة وذلك بإستخدام الدالة الحدية المقطعية (Sigmoid Function) المبينة سابقا.
- ٣ - طبقة المخرج (Output layer) وهى الطبقة التى تقع عند المخرج وتقوم عناصرها الحسابية بإجراء عملية الجمع الموزن (Weighted Sum) ووضع وتطبيق القيم الحدية للاثارة، وذلك بإستخدام الدالة الحدية المقطعية (Sigmoid Function) مثل الطبقة المخفية.



(١٧-٣ أ) - النموذج العام للشبكات المتعددة الطبقات

وحيث انه تم التعديل بإضافة طبقة مخفية بين طبقتى المدخل والمخرج واستخدام الدالة الحدية المقطعية (Sigmoid Function) والتي تحمل الخواص اللاخطية، لذلك يلزم إستحداث طريقة أخرى للتعليم، تكون قادرة على التفاعل بين الطبقات الثلاث مثل قاعدة دلتا العامة (Generalised Delta Rule) وقاعدة الانتشار الخلفى (Backpropagation Rule).

(١٧-١-٢) قاعدة دلتا المعممة للتعليم (Generalised Learning Delta Rule)

يطلق كذلك على هذه القاعدة قاعدة الانتشار الخلفى او العكسى ولقد اقترحها كل من (Rumelhart, McClelland & Williams) فى عام ١٩٨٦ ومن أهم الكتب التى عالجت هذا الموضوع هو الكتاب الذى تم نشره تحت عنوان المعالجة المتوازية الموزعة (Parallel Distributed Processing). وتعتبر قاعدة التعليم هذه معقدة بعض الشيء اذا قورنت بطريقة التعليم البسيطة للنيرون فاذا فرضنا ان هناك شبكة عصبية متعددة الطبقات وتحتوى على الثلاثة طبقات السابق تعريفها، ونريد لهذه الشبكة ان تتعلم للتعرف على شكل معين او بصمة معينة عند مدخل الشبكة لتعطى خرجاً معيناً عند المخرج يكون الدليل على نجاح عملية التعرف، وفيما يلى شرح لخطوات هذه الطريقة :

١ - نبدأ بتعريض الشكل المعين او البصمة عند مدخل الشبكة الغير مدربه حيث تقوم الشبكة

بعمل الحسابات اللازمة للجمع الموزن وتطبيق الدالة الحدية وحساب قيمة الخرج وتبعاً لذلك فإننا سوف نحصل على قيم عشوائية للخروج من الشبكة.

٢ - من الطبيعي ان نقوم بحساب دالة الخطأ بين قيم الخرج العشوائية التي تم الحصول عليها والتي تمثل الخرج الحقيقي في الخطوة الاولى وقيم الخرج المعين المطلوب.

٣ - لتقليل قيم دالة الخطأ فإننا نقوم بتعديل الاوزان في طبقة الخرج اولافى اتجاه تقليل الخطأ، ثم نشر او نقل هذا الخطأ الى الطبقة السابقة وتعديل الاوزان عند مداخل هذه الطبقة، وحساب قيم الخرج مرة ثانية للشبكة فى الوضع الجديد وحساب دالة الخطأ مرة اخرى مستخدماً الخرج الجديد والخرج المطلوب، وتعديل الاوزان مرة اخرى فى طبقة الخرج ثم نقل الخطأ الجديد الى الطبقة التي قبلها وهكذا .

٤ - يتم تكرار التعليم عدد مرات حتى تتلاشى دالة الخطأ ويصبح الخرج المحسوب هو الخرج المطلوب، وبذلك تتعلم الشبكة على التعرف الصحيح. وتعتبر عدد مرات التكرار هي المقياس لكى تتعلم الشبكة، كما يطلق على هذا النوع من التعليم للشبكة بالتعليم الموجه (Supervised Learning).

(١٧-١-٣) التمثيل الرياضى لقاعدة دلتا للتعليم
(Mathematical Representation of Learning Delta Rule)

نفرض ان هناك بصمة (Pattern P) شكل (١٧-٣ب) وان هناك شبكة ثلاثية الطبقات مثلاً ونفرض كذلك مايلى:

١- ان العناصر (i), (j) تمثل عناصر داخل الشبكة.

٢ - ان الوزن بين العنصرين هو W_{ij} .

٣ - ان الخرج الحقيقى عند مخرج العنصر (j) هو p_j .

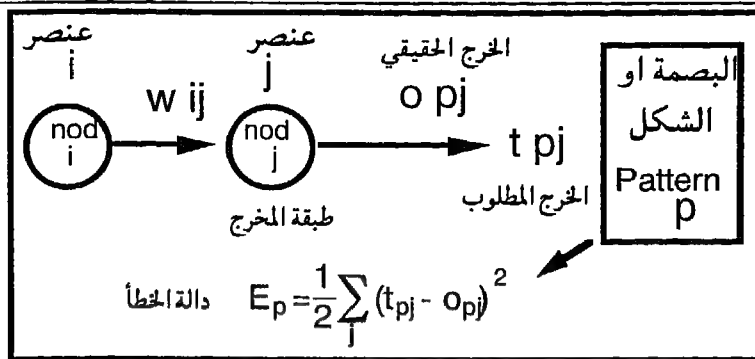
٤- ان الخرج المطلوب عند مخرج العنصر (j) هو t_j .

٥- ان الخطأ الكلي فى توصيف البصمة E_p .

ونفرض ان دالة الخطأ تتناسب مع مربع الفرق بين الخرج الحقيقى والخرج المطلوب فى الشكل الاتى:

$$E_p = \frac{1}{2} \sum_j (t_{pj} - o_{pj})^2 \quad \dots\dots\dots (١)$$

ويمكن حساب دالة التنشيط (net_{pj}) فى العنصر (j) من اجل التعرف على البصمة (P) والتي تعتبر مجموع حاصل ضرب الوزن فى قيمة الدخل كمايلى:



(١٧-٣) - التمثيل الرياضى لقاعدة دلتا المعممة

$$net_{pj} = \sum_i w_{ij} o_{pi} \quad \dots\dots\dots (٢)$$

حيث o_{pi} هو الخرج الفعلى من العنصر i .
ويكون الخرج الفعلى o_{pj} من العنصر (j) هو حاصل ضرب المجموع الموزن فى الدالة الحدية المقطعية (f_j) كمايلى:

$$o_{pj} = f_j (net_{pj}) \quad \dots\dots\dots (٣)$$

٦ - ويمكن كتابة التغير فى الخطأ الكلى بدلالة التغير فى الازان كمايلى:

$$\frac{\partial E_p}{\partial w_{ij}} = \frac{\partial E_p}{\partial net_{pj}} \frac{\partial net_{pj}}{\partial w_{ij}} \quad \dots\dots\dots (٤)$$

ويمكن تعديل الحد الثانى من المعادلة (٤) والذى يشمل التغير فى الدالة الحدية المقطعية بدلالة التغير فى الازان ليصبح الخرج الفعلى o_{pi} من العنصر (i) كمايلى:

$$\begin{aligned} \frac{\partial net_{pj}}{\partial w_{ij}} &= \frac{\partial}{\partial w_{ij}} \sum_k w_{kj} o_{pk} \\ &= \sum_k \frac{\partial w_{kj}}{\partial w_{ij}} o_{pk} = o_{pi} \quad \dots\dots\dots (٥) \end{aligned}$$

حيث ان القيمة

$$\frac{\partial w_{kj}}{\partial w_{ij}} = 0$$

فى جميع الاحوال ما عدا عند $k = i$ فإن قيمتها تساوى الوحدة δ_{pj}

٨ - فإذا فرضنا ان التغير فى الخطأ كدالة فى التغير فى مداخل الشبكة للعنصر j هو كمايلى:

$$-\frac{\partial E_p}{\partial \text{net}_{pj}} = \delta_{pj} \quad \dots\dots\dots (٦)$$

وبالتعويض فى المعادلة (٤) يصبح التغير فى الخطأ الكلى بدلالة التغير فى الازان كمايلى:

$$-\frac{\partial E_p}{\partial w_{ij}} = \delta_{pj} o_{pi} \quad \dots\dots\dots (٧)$$

وبصيح تقليل الخطأ E_p هو فى الحقيقة محاولة تقليل الازان بقيم تناسب مع $\delta_{pj} o_{pi}$ والتى يمكن كتابتها فى الشكل :

$$\Delta_p w_{ij} = \eta \delta_{pj} o_{pi} \quad \dots\dots\dots (٨)$$

حيث يعتبر المعامل $0 \leq \eta \leq 1$ هو الذى يتحكم فى معدل الانضباط.
٩ - ولتحديد قيمة التغير δ_{pj} فى دالة الخطأ لكل عنصر يمكننا تحديد كيفية تعديل دالة الخطأ الكلية E_p وذلك بكتابة مايلى:

$$\delta_{pj} = -\frac{\partial E_p}{\partial \text{net}_{pj}} = -\frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial \text{net}_{pj}} \quad \dots\dots\dots (٩)$$

حيث يمثل الجزء الاول من المعادلة (٩) من الطرف الايمن مايلى:

$$\frac{\partial E_p}{\partial o_{pj}} = -(t_{pj} - o_{pj}) \quad \dots\dots\dots (١٠)$$

ويمثل الجزء الثانى من المعادلة (٩) من الطرف الايمن مايلى:

$$\frac{\partial o_{pj}}{\partial \text{net}_{pj}} = f'_j(\text{net}_{pj}) \quad \dots\dots\dots (١١)$$

وبذلك تصبح قيمة دلتا δ_{pj} كمايلى:

$$\delta_{pj} = f'_j(\text{net}_{pj}) (t_{pj} - o_{pj}) \quad \dots\dots\dots (١٢)$$

والتي تعطى مقدار التغير بين القيم المستهدفة (Target) والقيم الفعلية (Actual) للخروج عند الطبقة الاخيرة (طبقة المخرج)، ومن الملاحظ انه يمكن حساب هذه الكمية

ببساطة حيث ان القيم الفعلية والمستهدفة معروفة، وليس هذا الحال بالنسبة للطبقة المختفية حيث يختلف الامر من عدم معرفة القيم المستهدفة.

١٠- اذا فرضنا ان العنصر j موجود فى الطبقة المختفية وليس فى طبقة المخرج، فإنه يمكن كتابة مايلى:

$$\begin{aligned} \frac{\partial E_p}{\partial o_{pj}} &= \sum_k \frac{\partial E_p}{\partial net_{pk}} \frac{\partial net_{pk}}{\partial o_{pj}} \\ &= \sum_k \frac{\partial E_p}{\partial net_{pk}} \frac{\partial}{\partial o_{pj}} \sum_i w_{ik} o_{pi} \dots\dots\dots (١٣) \\ &= - \sum_k \delta_{pk} w_{jk} \dots\dots\dots (١٤) \end{aligned}$$

وباستخدام المعادلات (٢)، (٦)، (٥)، وبالتعويض فى (٩) بإستخدام (١٤) فإنه يمكن الحصول على مايلى:

$$\delta_{pj} = f'_j(net_{pj}) \sum_k \delta_{pk} w_{jk} \dots\dots\dots (١٥)$$

والتي تعطى مقدار التغير فى دالة الخطأ بالنسبة للتغير الحادث فى الاوزان للشبكة δ_{pj} ومعتمدة على الخطأ فى الطبقة السابقة δ_{pk} والتي تساعد على تغيير دالة الخطأ فى الاتجاه الى تقليله وذلك بالترتيب الآتى:

١ - يجرى حساب مقدار الخطأ δ_{pj} لعناصر الطبقة الاخيرة (طبقة المخرج) اولا وذلك بإستخدام المعادلة رقم (١٢) فى الشكل :

$$\delta_{pj} = f'_j(net_{pj}) (t_{pj} - o_{pj}) \dots\dots\dots (١٦)$$

٢ - استخدام ظاهرة الانتشار العكسى (Back Propagation) او الانتشار الى الخلف او المرتد) وذلك بتمرير الخطأ الى عنصر الطبقة المختفية او الطبقة التى تلى طبقة المخرج وحساب الاوزان وتغيير القيم لها، وذلك بإستخدام المعادلة رقم (١٥) فى الشكل:

$$\delta_{pj} = f'_j(net_{pj}) \sum_k \delta_{pk} w_{jk} \dots\dots\dots (١٧)$$

٣ - وبإستخدام هاتين المعادلتين يمكن تعليم (Train) الشبكة المتعددة الطبقات تحت مظلة الانتشار الى الخلف (Back Propagation).

ومن اهم فوائد استخدام الدالة الحدية المقطعية (Sigmoid Threshold Function) فى الشكل اللاخطى (Non-linear) الاتى:

$$f(\text{net}) = \frac{1}{1 + e^{-k \text{ net}}} \quad \dots\dots\dots(18)$$

كمايلى:

١- ان مدى هذه الدالة ينحصر بين الصفر والواحد الصحيح $0 < f(\text{net}) < 1$ ، وان المعامل k هو ثابت موجب ويقوم بالتحكم فى تدرج إنحناء الدالة الحدية، حيث يكون الإنحناء كبيراً عند القيم الصغرى للثابت (k) ، وتأخذ الدالة شكل دالة (Heaviside) عند القيم الكبيرة له.

٢- يمكن تعريف الثابت k على انه المتحكم التلقائى فى مكسب الاشارة (Automatic Gain Control) حيث يمكن التحكم فى التغيير عندما تكون الاشارة الواقعة على الخط المائل للدالة صغيرة ويكون المكسب كبيراً، وعلى العكس عندما تكون الاشارة كبيرة فإن التغيير يكون قليلاً، وهذا يعنى ان الشبكة تكون مناسبة لقيم المداخل الكبيرة وتظل صالحة للقيم الصغرى.

٣ - ان هذه الدالة سهلة التفاضل وبذلك تكون مناسبة للاستخدام فى شبكات الانتشار الخلفى او المرتد (Back Propagation)، فإذا فرضنا ان قيمة الخرج من احد العناصر هو Op_j كمايلى:

$$o_{pj} = f(\text{net}) = \frac{1}{1 + e^{-k \text{ net}}} \quad \dots\dots\dots(19)$$

ويكون المشتق التفاضلى الاول دالة فى قيم الخرج كمايلى:

$$\begin{aligned} f'(\text{net}) &= \frac{k e^{-k \text{ net}}}{1 + (e^{-k \text{ net}})^2} \\ &= k f(\text{net}) (1 - f(\text{net})) \\ &= k f o_{pj} (1 - o_{pj}) \quad \dots\dots\dots(20) \end{aligned}$$

(١٧-١-٤) خوارزم التعليم للنموذج متعدد الطبقات

(The Multilayer Model Learning Algorithm)

لبناء خوارزم التعلم للشبكات العصبية المتعددة الطبقات باستخدام قاعدة الانتشار العكسي (Back Propagation) فإنه يلزم الاستعانة بدالة حدية قابلة للتفاضل، وتأتي في المقدمة الدالة الحدية المقطعية المذكورة سابقاً (Sigmoid Threshold Function) وتشتمل خطوات البناء ما يلي:

١ - بدء التنشيط (Initialise) للأوزان (Weights) والقيم الحدية (Thresholds) ويتم

ذلك بإختيار قيم عشوائية (Random) صغيرة للأوزان والقيم الحدية.

٢ - تحديد قيم المدخل (Input) والقيم المطلوبه عند المخرج (Desired Output) كما يلي:

أ - ضع قيم المداخل x_p في الشكل الاتي:

$$X_p = x_1 + x_2 + x_3 + \dots + x_{n-1} \quad \dots\dots\dots (١)$$

حيث الرقم (n) هو عدد العقد في طبقة المدخل (Input Nodes).

ب - ضع قيمة w_0 لتمثل القيمة الانحيازية (Bias) $\theta -$.

ج - ضع قيمة x_0 لتكون دائماً تساوى الواحد الصحيح.

د - ضع قيم المخرج المطلوب او المستهدف T_p في الشكل الاتي:

$$T_p = t_0 + t_1 + t_2 + t_3 + \dots + t_{m-1} \quad \dots\dots\dots (٢)$$

حيث الرقم (m) هو عدد العقد في طبقة المخرج (Output nodes).

هـ - ضع جميع عناصر T_p مساوية للصفر ماعدا عنصر واحد يساوى الواحد الصحيح والذي يقابل المطلوب.

٣ - احسب المخرج الفعلى او الحقيقي (Actual) لكل طبقة وذلك كما يلي:

أ - يتم حساب قيمة المخرج y_{pj} للطبقة الاولى من ناحية المدخل في الشكل:

$$y_{pj} = f \left[\sum_{i=0}^{n-1} w_i x_i \right] \quad \dots\dots\dots (٣)$$

ب - يتم إتخاذ قيمة المخرج y_{pj} لهذه الطبقة كقيم لمدخل الطبقة التي تليها، ويتم

تكرار العملية الى ان نصل الى الطبقة الاخيرة حيث تكون قيمة المخرج هو o_{pj} .

٤ - ضبط الاوزان (Weights Adaptation)، ويتم ذلك في الاتجاه العكسي

(من طبقة المخرج الى طبقة المدخل) وذلك كما يلي:

أ - يجرى بدء تعديل الاوزان بالشكل الاتي:

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_{pj} o_{pj} \quad \dots\dots\dots(4)$$

حيث $w_{ij}(t)$ تمثل الاوزان من العقدة او العنصر أ الى العنصر ج عند الزمن t .

و (η) هو معامل الكسب (Gain Term) و δ_{pj} هو مقدار الخطأ عند العنصر ج وبذلك يصبح الخطأ لعناصر طبقة المخرج مايلي:

$$\delta_{pj} = k o_{pj} (1 - o_{pj}) (t_{pj} - o_{pj}) \quad \dots\dots\dots(5)$$

وللطبقة المخفية كمايلي:

$$\delta_{pj} = k o_{pj} (1 - o_{pj}) + \sum_k \delta_{pk} w_{jk} \quad \dots\dots\dots(6)$$

(١٧-١-٥) وضع الحل لمشكلة الدائرة المنطقية (أو- المانعة)

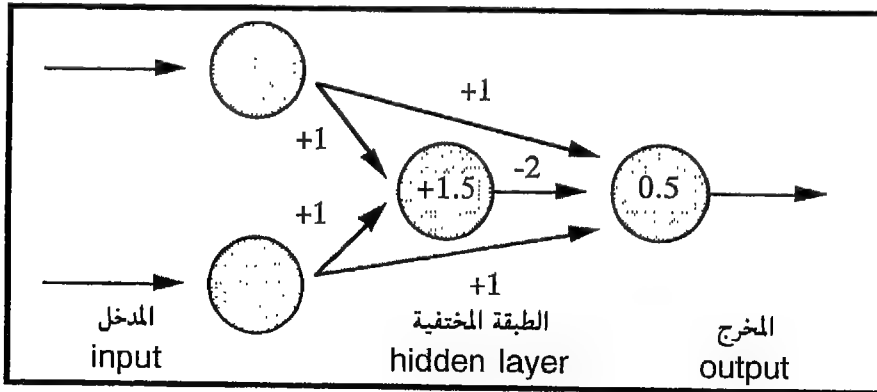
(Solution Of (XOR) Problem)

مما تم عرضه سابقاً من عدم قدرة الشبكة العصبية ذات الطبقة الواحدة من وضع الحل الصحيح لمشكلة الدائرة أو - المانعة (XOR) والذي يعبر عن حدود التطبيق للنيرون ذو الطبقة الواحدة، وفيمايلي يجرى إستخدام شبكة عصبية مكونة من ثلاث طبقات حيث تمثل الطبقة الوسطى الطبقة المخفية ومدخلين ومخرج واحد والمبينة فى شكل (١٧-٤) لوضع الحل لهذه المشكلة.

من الواضح بالشكل ان الاوزان مبينة على الاسهم والقيم الحدية مبينة داخل العناصر

الحسابية، ويمكن شرح عمل الشبكة كمايلي:

١ - اذا كانت قيم المداخل (0 0) اى ان عناصر المدخل غير نشطة (Off) وبذلك يصبح



شكل (١٧-٤) شبكة عصبية توضع الحل لمشكلة الدائرة أو- المانعة (XOR)

العنصر الحسابى للطبقة المختفية غير نشطاً كذلك وبالتالي فإن عنصر طبقة المخرج لا يصبح نشطاً كذلك وتكون قيمة المخرج صفراً.

٢ - إذا كانت قيم المداخل (0 1) أى ان احد عناصر المدخل غير نشطة (off) وبذلك يصبح العنصر الحسابى للطبقة المختفية غير نشطاً ويصبح عنصر طبقة المخرج نشطاً (on) وذلك بسبب وجود الواحد عند مدخلها والذي يفوق القيمة الحدية (٠.5) وتكون قيمة المخرج (1).

٣ - إذا كانت قيم المداخل (1 0) أى انه يتم تبادل وضع عناصر المدخل وتكرر الخطوة السابقة، وبذلك يصبح العنصر الحسابى للطبقة المختفية غير نشطاً ويصبح عنصر طبقة المخرج نشطاً (on) وذلك بسبب واحد عند مدخلها والذي يفوق القيمة الحدية (٠.5) وتكون قيمة المخرج (1) .

٤ - إذا كانت قيم المداخل (1 1) أى ان عناصر المدخل نشطة (on) وبذلك يصبح العنصر الحسابى للطبقة المختفية نشطاً كذلك وذلك بسبب وجود المجموع (2) للاوزان عند مدخله والذي يفوق القيمة الحدية لعنصر الطبقة المختفية (1.5) ويكون المخرج له يساوى (1) وبالتالي فإن عنصر طبقة المخرج لا يصبح نشطاً وتكون قيمة المخرج صفراً وذلك بسبب وجود المجموع الموزن الآتى:

$$1 \times 1 + 1 \times 1 + 1 \times (-2) = 0$$

عند مدخله والتي تقل عن القيمة الحدية للعنصر (٠.5) .

ومن الملاحظ ان الطبقة المختفية تعمل مثل كاشف للملامح او السمات (Featur Detector) حيث يكتشف حالة النشاط لعنصرى طبقة المدخل.

(١٧-٢) سلوك الشبكات المتعددة الطبقات

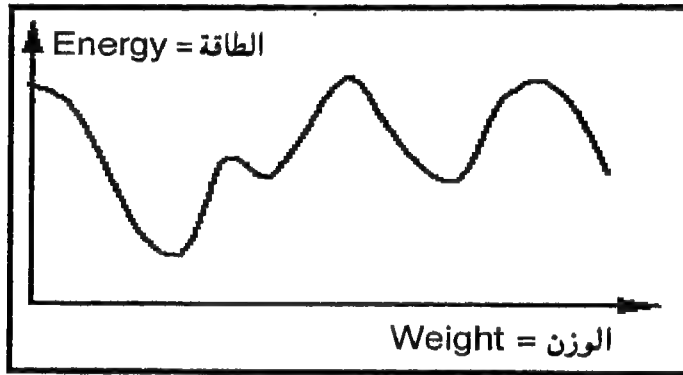
(Behaviour Of Multilayer Network)

فيما سبق تم شرح قاعدة دلتا المعممة رياضياً وإبراز كيف يمكن نقل الخطأ من المخرج الى الطبقة المختفية وضبط الاوزان وتعليم الشبكة عدة مرات ثم تم شرح كيف يمكن للشبكة المتعددة الطبقات من وضع الحل للدائرة (او) المانعة، وللوصول الى تفهم واضح لسلوك الدائرة فإننا سوف نقوم بتعديل تعريف دالة الخطأ لتصبح دالة للطاقة (Energy) والتي تمثل مقدار طاقة الفرق بين قيم المخرج الفعلى والمطلوب كمايلي:

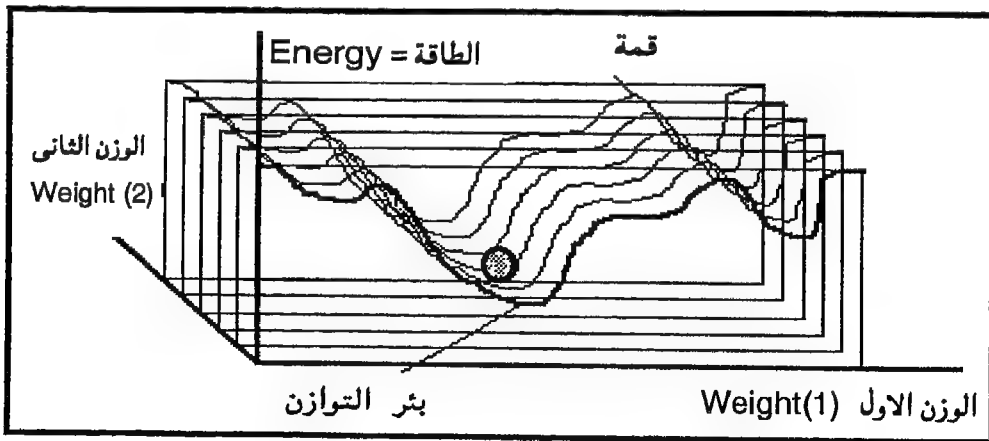
$$\text{Energy} = \text{الطاقة} = E_p = \frac{1}{2} \sum_j (t_{pj} - o_{pj})^2$$

وتصبح دالة الطاقة معتمدة على الاوزان (Weights) بين عناصر الشبكة وقيم المداخل (Inputs) للشبكة، ويمكن تمثيل منحني الطاقة بيانيا كما هو مبين في شكل (١٧-٥) وذلك اذا اعتبرنا ان الشبكة ثابتة ولايتغير فيها الاوزن واحد فقط.

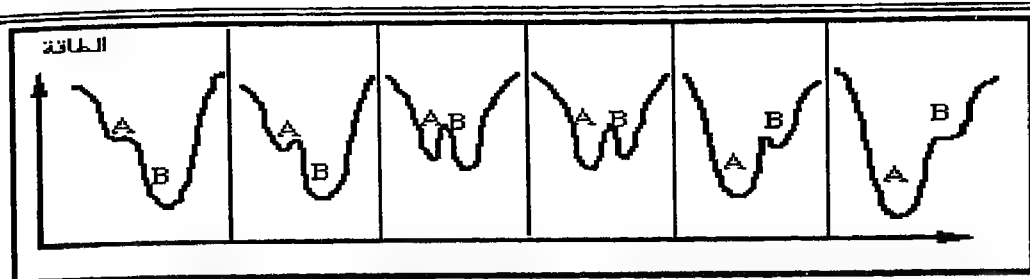
واذا فرضنا اننا سوف نقوم بتغيير وزن فقط، فان ذلك يمكن تمثيله برسم في فراغ ثلاثي الابعاد (3-D) حيث يمثل كل وزن احد الاحداثيات ويحتوى على القمم (Hills) والقيعان او الوديان (Valleys) ويصبح اكبر انخفاض في احد الوديان هو المقابل لاقبل قيمة لدالة الطاقة والذي يعرف ببئر التوازن الذي يشير الى حالة التطابق بين الخرج المطلوب والخرج الفعلي كما هو مبين في شكل (١٧-٦).



شكل (١٧-٥) التغير في دالة الطاقة طبقاً للتغير في احد الاوزان



شكل (١٧-٦) تمثيل دالة الطاقة في ثلاثة ابعاد طبقاً للتغير في وزنين فقط



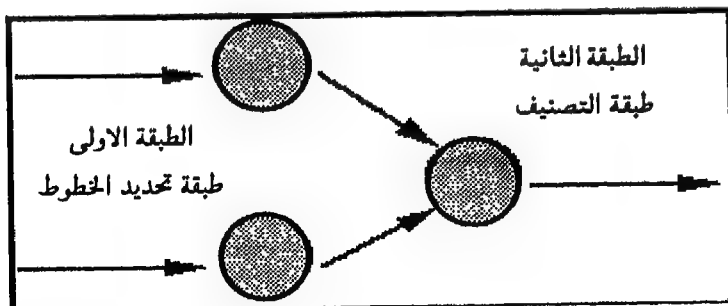
شكل (١٧-٧) تعديل الوزن في اتجاه التعرف على الشكل (A)

ومن المعروف ان الشبكات الحقيقية تتطلب تعديل الكثير من الاوزان وتبعاً لذلك تتحول دالة الطاقة الى شكل فراغى متعدد الابعاد يصعب حتى التفكير في قمته، ولذلك يمكن القول بان السطح المتعدد الابعاد سوف يحتوى على القمم التى تقابل الطاقات العظى والتى تبعد كل البعد عن الحل المطلوب والوديان والتى تقابل قيم الطاقات الصغرى التى تكون اصغرها هى النقطة المطلوبة، ومن هذا المنطلق فإن قاعدة دلتا المعممة تبحث عن اقل مستوى للطاقة وذلك باستخدام طريقة هبوط الميل (Gradient Descent). وبين شكل (١٧-٧) تصور مبسط فى بعد واحد كيف يجرى تعديل الوزن فى إتجاه الوصول الى التعرف على الحرف (A).

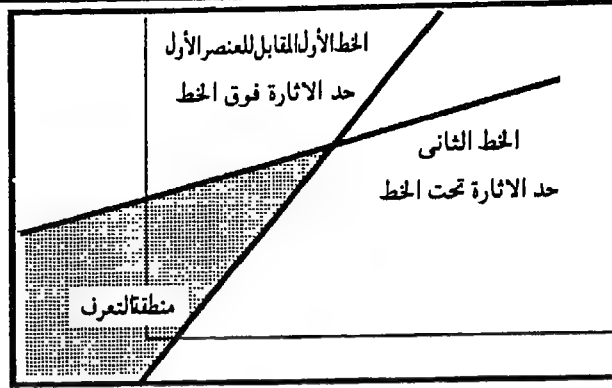
(١٧-٣) الشبكات المتعددة الطبقات كمصنفات

(Multilayer Network As Classifiers)

كما رأينا سابقاً ان الشبكة المكونة من عنصر واحد (نيرون واحد) والذي يقوم بعملية الجمع الموزن وتطبيق الدالة الحدية للاستشارة تكون غير قادرة على وضع الحل لدائرة (او) المانعة حيث ان الاخيرة لا يمكن فصلها بخط مستقيم واحد (Linearly Inseparable). وكما رأينا كذلك ان الشبكة المكونة من طبقتين قادرة على الفصل ووضع الحل لدائرة (او) المانعة. ولايضاح قدرة الشبكات العصبية على التصنيف نفرض شبكة عصبية مكونة من ثلاثة عناصر حسابية والمبينة فى شكل (١٧-٨).



شكل (١٧-٨) شبكة الثلاث عناصر للتعرف والتصنيف



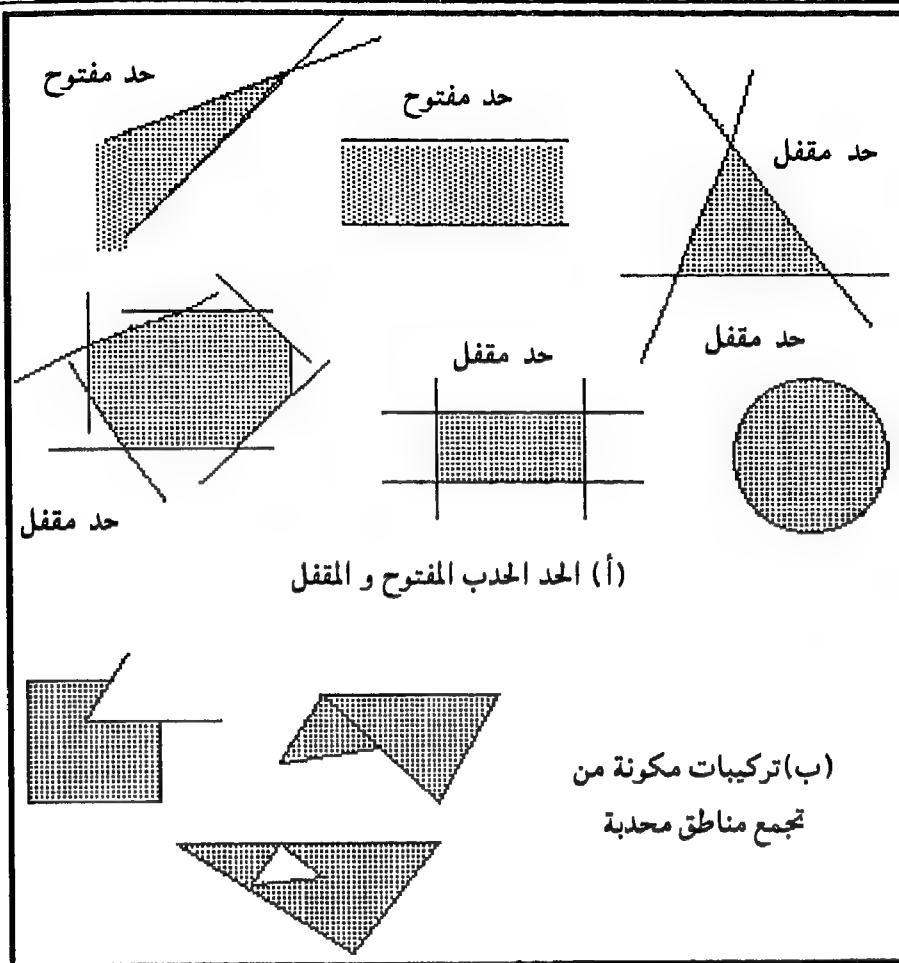
شكل (١٧-٩) منطقة التعرف (او اتخاذ القرار) فوق الخط الاول وتحت الخط الثاني

ومن الواضح ان العنصر فى الطبقة الثانية لا يعطى خرجاً يساوى الوحدة الا فى حالة واحدة فقط وذلك عندما يكون الخرج من العنصرين فى الطبقة الاولى مساو للواحد الصحيح، اى ان هذه المجموعة تقوم بعمل دائرة (و) المنطقية (Logical AND) حيث ان كل عنصر فى الطبقة الاولى يقوم بتمثيل احد الخطوط فى الفراغ الذى يمثل حيز البصمة او الشكل ويقوم العنصر الثالث فى الطبقة الثانية بعمل التصنيف (Classification) الذى يعتمد على التجميع للخطين كما هو موضح بشكل (١٧-٩) حيث تكون منطقة التعرف هى المنطقة المحصورة فوق الخط الاول وتحت الخط الثانى.

(١٧-٣-١) الحدود والمناطق المحدبة (Convex Hulls and Convex Regions)

تتكون الحدود والمناطق المحدبة فى الشبكات اذا زاد عدد العناصر الحسابية فى طبقة المدخل عن اثنين حيث يقوم كل عنصر بتمثيل خط مستقيم وتتقاطع هذه الخطوط لتعطى مناطق وحدود محدبة كما هو مبين فى شكل (١٧-١٠) ويعرف الحد المحدب (Convex Hulls) بانه التقاطع للخطوط بحيث لا يمر الخط المستقيم داخل منطقة التقاطع وبينما تعرف المناطق المحدبة (Convex Regions) بانها المناطق التى تتكون من تقاطع الحدود المحدبة، وفى كثير من الاحوال تتكون تركيبات من تجمع اكثر من منطقة محدبة ويمكن ابراز الحقائق الاتية :

- ١ - إن اضافة اى عنصر فى الطبقة الاولى سوف يؤدى الى زيادة عدد الاضلاع للحد المحدب وبذلك تتساوى عدد الاضلاع مع عدد العناصر الحسابية فى الطبقة الاولى.
- ٢ - عند اضافة طبقة ثانية الى الطبقة الاولى فإن عناصر هذه الطبقة سوف تستقبل حدود محدبة التى يمكن ان تقاطع او تنفصل او تقع فوق بعضها مكونة لتركيبات مختلفة.

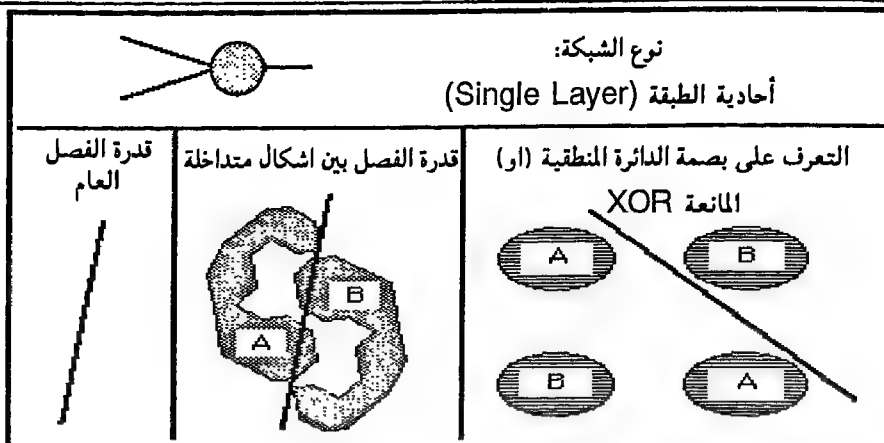


شكل (١٧-١٠) الحدود والمناطق المحدبة والمعتمدة على عدد العناصر في الطبقة الاولى

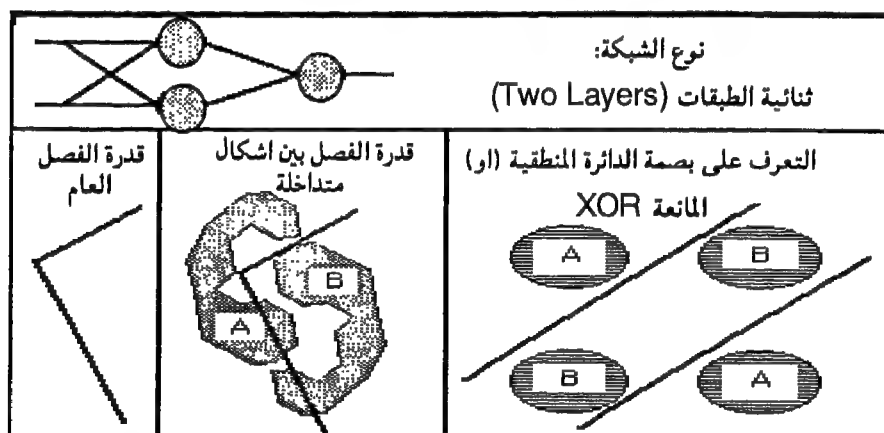
٣ - عند إضافة طبقة ثالثة فإنه يمكن الحصول على تركيبات أكثر تعقيداً حيث تعتمد درجة التعقيد على عدد العناصر الحسابية في الشبكة والتي تعني أننا لا نحتاج إلى أكثر من ثلاث طبقات للتعرف أو الفصل لأي بصمات أو أشكال مهما كانت درجة التعقيد، وهذا الفرض البسيط هو في الحقيقة أساس نظرية كولموجوروف (Kolmogorov Theorem) التي تنص على مايلي:

” أن الشبكة ذات الثلاث طبقات قادرة على وضع الحل من التعرف أو الفصل أو التصنيف لأي مشكلة مهما بلغت من التعقيد ”.

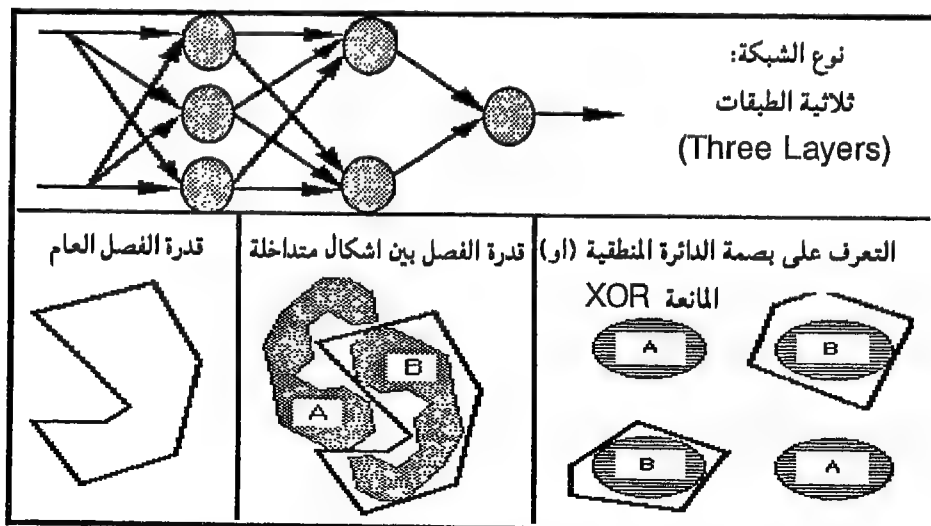
ويمكن شرح ذلك في الشكل (١٧-١١) للشبكة أحادية الطبقة والشكل (١٧-١٢) للشبكة الثنائية للطبقة والشكل (١٧-١٣) للشبكة ثلاثية الطبقات.



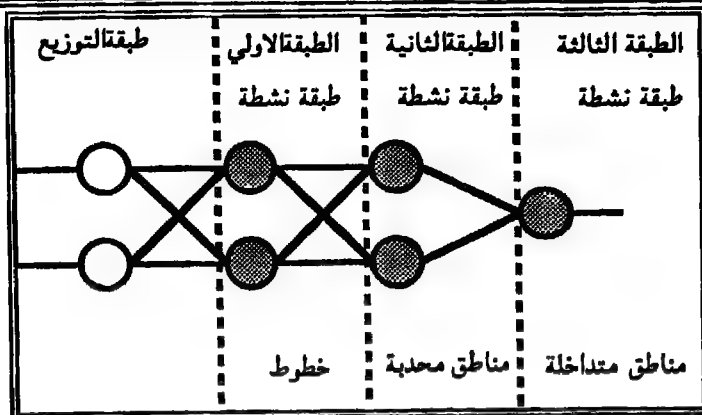
شكل (١٧-١١) قدرة الفصل والتصنيف للشبكة احادية الطبقة



شكل (١٧-١٢) قدرة الفصل والتصنيف للشبكة ثنائية الطبقات



شكل (١٧-١٣) قدرة الفصل والتصنيف للشبكة ثلاثية الطبقات



شكل (١٧-١٤) قدرة الفصل للشبكة ثلاثية الطبقات

(١٧-٣-٢) تحديد طبيعة العمل للطبقات (Function Determination of Layers)

من الملاحظ انه يوجد كثير من عدم الوضوح بالنسبة لبعض ما تم نشره في بعض الكتب عن الشبكة الثلاثية الطبقات والتي تتركب من الطبقات الاتية كما في شكل (١٧-١٤) :

- ١ - طبقة التوزيع والتي تشتمل على عدد من العناصر يحدد عددها عدد الخطوط في الحل النهائي وتقوم بتوزيع القيم للمداخل على عناصر الطبقة التي تليها.
- ٢ - الطبقة الاولى وهي طبقة نشطة تقوم بعمل الجمع الموزن وتحدد طبيعة الخطوط.
- ٣ - الطبقة الثانية وهي طبقة نشطة تقوم بعمل الجمع الموزن وتحدد طبيعة المناطق المحدبة عند مخرجها.
- ٤ - الطبقة الثالثة وهي طبقة نشطة تقوم بعمل الجمع الموزن وتحدد طبيعة المناطق المتداخلة عند مخرجها.

(١٧-٤) خاصية التعميم (Generalisation Property)

تعتبر خاصية التعميم من اهم السمات الاساسية للشبكات العصبية المتعددة الطبقات حيث يمكن للشبكة ان تقوم بتصنيف بصمات لم تدرب عليها من قبل، وتوصف هذه الخاصية بانها قدرة الشبكة على اكتشاف الملامح (Features) للشكل او البصمة المدخلة والمطابقة او القربة من الملامح التي تعلمت عليها الشبكة من بصمات اخرى مسجلة او مشفرة بداخل هذه الشبكات، والتي تعنى ان التعرف والتصنيف يتم باستخدام ملامح مشتركة في البصمة او الشكل المدخل وملامح ماثلة معرفة لبصمات اخرى. ومن فوائد خاصية التعميم مايلي :

- ١- رفع كفاءة طريقة التعليم للشبكة باستخدام الأمثلة (Learning by Examples) حيث يجرى تعليم الشبكة علي ملامح مجموعة من البصمات ثم التعرف على بصمة اخرى لم يتم التعلم عليها باستخدام الملامح المعروفة من خلال خاصية التعميم.

- ٢- تساعد هذه الخاصية على التخلص من التشويش او الضوضاء الالكترونية (Noise) حيث يمكن التعرف على بصمات مشوشة بكمية كبيرة من الضوضاء.
- ٣- تؤدي هذه الخاصية الي ان تتفوق الشبكات العصبية فى تفسير البصمات التى تحتوى على ملامح معرفة سابقاً (Interpolation) من قبل الشبكة ولايمكنها التفسير المسبق (Extrapolation) لبصمات لا تحتوى على اى من الملامح المعرفة من قبل.

(١٧-٥) التجاوز عن الخطأ (Fault Tolerance)

حيث ان طريقة العمل للشبكات العصبية تعتمد على المعالجة الموزعة المتوازية (Distributed Parallel Processing) والتى تسمح للشبكات العصبية بان تقوم بالعمل تحت نسبة كبيرة من الخطأ المسموح به، ويجرى تفسير ذلك لطبيعة عمل العناصر الحساسة للشبكة التى تعمل على التوازي لتعطى خرجاً معيناً. فإذا فرضنا مثلاً ان شبكة ما تحتوى على مائة عنصر حسابى فإن الخرج يعتمد على عمل هذه المائة عنصر، والذي لا يتأثر فى حالة غياب او تلف بعض منها او نسبة كبيرة منها عند التعرف او اجراء التصنيف، وطبقاً لذلك فإن هذه الشبكات تكون قادرة على التعلم والتعرف فى وجود نسب كبيرة من الضوضاء او التشويش.

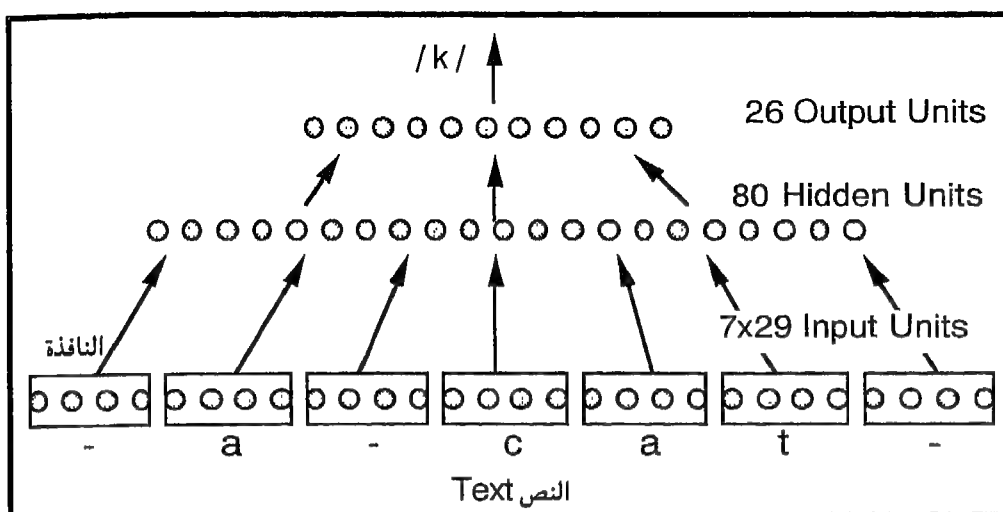
(١٧-٦) بعض تطبيقات الشبكات متعددة الطبقات

(١٧-٦-١) الشبكة الناطقة (NETTALK)

تعتبر الشبكة الناطقة من أشهر التطبيقات للشبكات العصبية الاصطناعية المتعددة الطبقات والتى تم تدريبها لنطق النصوص المكتوبة باللغة الانجليزية والتى تم نشرها لأول مرة فى عام ١٩٨٧ من قبل سيجنوسكى وروزنبرج (Sejnowski & Rosenberg).

تتركب الشبكة من (٢٠٣) عنصراً فى طبقة الادخال و(٨٠) عنصراً فى الطبقة المخفية، علاوة على (٢٦) عنصراً فى طبقة المخرج بواقع عنصر لكل فونم (Phoneme) حيث يمثل الاخير الوحدة الصوتية التى يتكون منها الحديث كما هو مبين فى شكل (١٧-١٥).

لكى تقوم الشبكة بنطق الكلمات الانجليزية المكتوبة، فلقد تم استخدام نافذة (Window) عرضها سبعة حروف بحيث تقوم الشبكة بالتعلم على نطق الحرف الاوسط (مثل الحرف C كما هو واضح فى الشكل، حيث ان النطق للحرف الاوسط فى النافذة يعتمد على الحرف الذى قبله والحرف الذى بعده، وبذلك تتعلم الشبكة النطق السليم للحرف طبقاً لموقعه فى الكلمة مثل



شكل (١٧-١٥) الشبكة العصبية الناطقة (NETTALK)

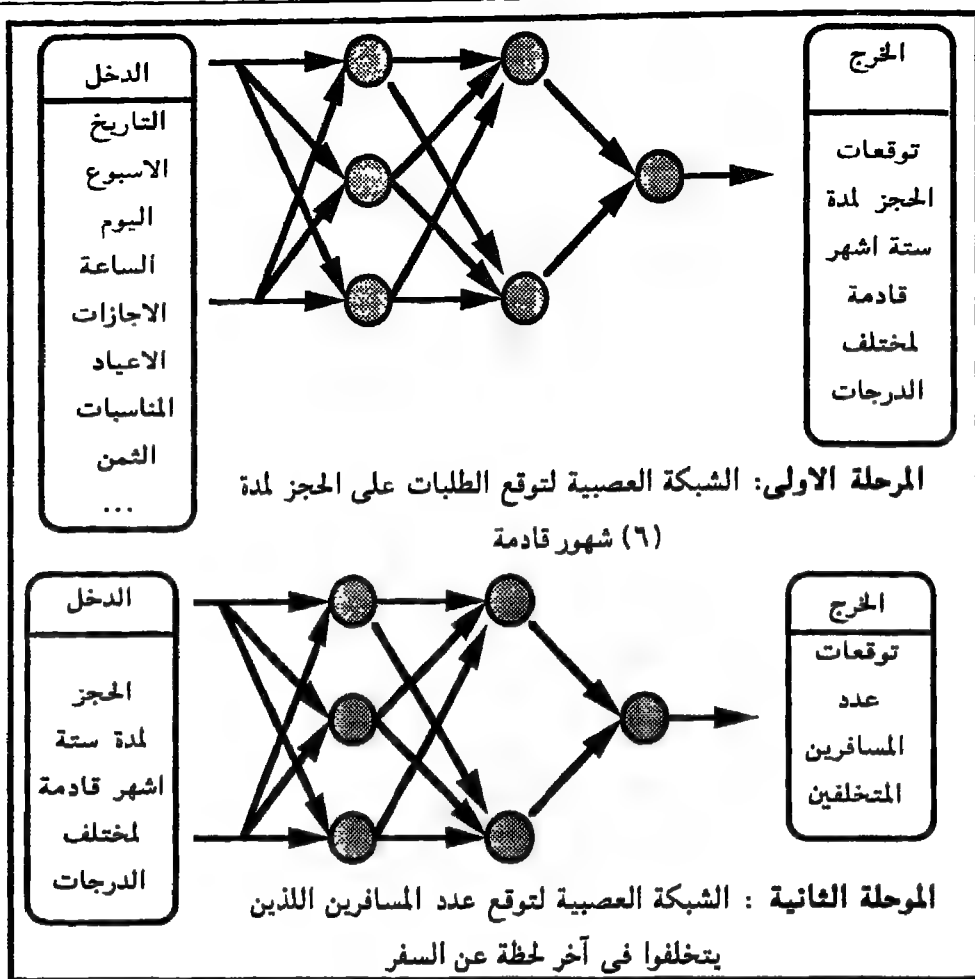
الحرف (c) المتبوع بالحرف (a) والذي يتم نطقه فى شكل حرف (k) ولقد أمكن تدريب الشبكة للنطق الصحيح للكلمات الانجليزية بكفاءة تساوى ٩٠٪. عند تعميم الاستخدام بمحاولة نطق كلمات من القاموس لم تتعلم عليها الشبكة قبل ذلك وصلت الكفاءة ما بين ٨٠٪ و ٨٧٪. وعند اضافة تشويش عشوائى تمكنت الشبكة من مقاومته بشكل كبير.

(١٧-٦-٢) المخطط التسويقي لشركات الطيران (Airline Marketing Tactican)

من اهم المشاكل التسويقية التى تقابل شركات الطيران هو تخلف وعدم وصول بعض المسافرين الى ارض المطار قبل موعد الاقلاع، وبذلك تطير الطائرة وبها بعض الأماكن الشاغرة والتى تعتبر من الناحية المالية خسارة على الشركة حيث انها مقاعد بدون عائد مادي، وذلك يتطلب معرفة مسبقة لعدد المسافرين الذين يتخلفون قبل الاقلاع بوقت كاف حتى يمكن التصرف باعادة الحجز لأماكنهم بالطائرة. يتكون النظام من مرحلتين كما هو مبين فى شكل (١٧-١٦) مايلى:

١- المرحلة الاولى: الشبكة العصبية لتوقع الطلبات على الحجز لمدة (٦) شهور قادمة والتى تقوم بالتعلم والتدرب على قوائم حقيقية للحجز تم تسجيلها على مدى العام الكامل مع الاخذ فى الاعتبار التاريخ والاسبوع واليوم والساعة والاجازات والاعیاد والمناسبات والثلثين وما الى ذلك ثم تقوم الشبكة المدربة بعد ذلك باجراء توقع لطلبات الحجز لمدة (٦) شهور قادمة.

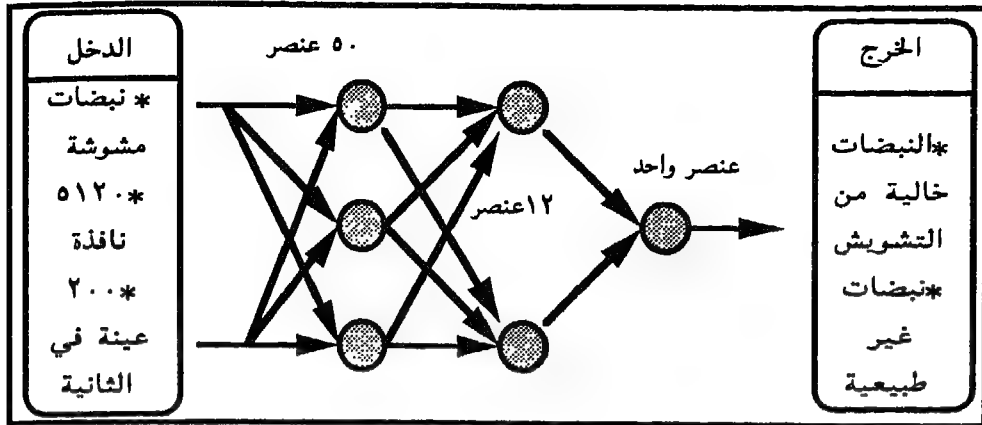
٢ - المرحلة الثانية: الشبكة العصبية لتوقع عدد المسافرين اللذين يتخلفوا فى آخر



شكل (١٦-١٧) مراحل عمل الشبكة لتخطيط التسويق لشركات الطيران (AMI)

لحظة عن السفر وذلك بعد ان تتدرب الشبكة على قوائم المتخلفين وأسباب التخلف وكيفية تم التصرف في هذه المشكلة، ثم تقوم بالتوقع بعد ذلك طبقاً للقيم المدخلة لقوائم الحجز. وتظل الشبكة في تحديث التعلم والتدريب لادخال المتغيرات التي تحدث في مجال الحجز والتسويق ولقد لاقى الشبكة نجاحاً في التطبيق العملي بشركات الطيران.

(١٧-٦-٣) تنقية رسم القلب الكهربى من التشويش (ECG Noise Filtering) من المعروف انه يوجد الكثير من الضوضاء الكهربائية (Noise) والتشويش عند اجراء الرسم الكهربى لتسجيل نبضات القلب مما يجعل من الصعوبة التحقق الكامل من شكل النبضة الاصلية، ولذلك قامت شركة (Hecht-Neilsen Neurocomputer) الامريكية بتطوير نظام شبكة



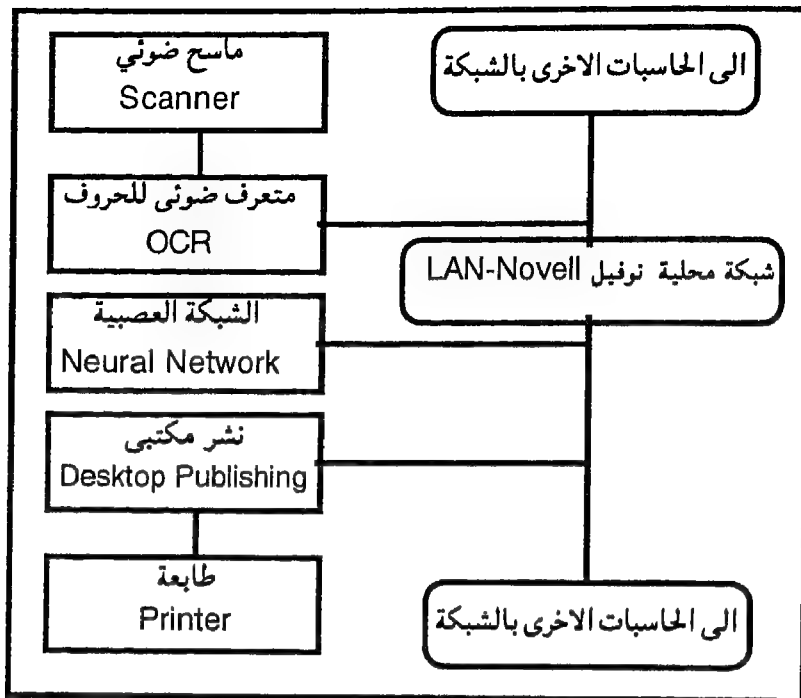
شكل (١٧-١٧) - تنقية رسم القلب الكهربائي من التشويش

عصبية صناعية تقوم بترشيح وتنقية النبضات من التشويش والحصول على نبضات نظيفة. وتحتوي الشبكة على (٥٠) عنصراً لطبقة المدخل و(١٢) عنصراً للطبقة المخفية وعنصر واحد لطبقة الإخراج، وتستقبل الشبكة (٥٠) جزء من النبضة عند المدخل (نافذة زمنية) لتعطي خرجاً خالٍ من الضوضاء عند المخرج يقابل منتصف النافذة الزمنية، ويعني ذلك أن الشبكة تأخذ في الاعتبار الجزء الذي يأتي قبل والجزء الذي يأتي بعد هذا الجزء، وذلك للتأكد من تسلسل النبضة والتخلص من الضوضاء. ولقد تم استخدام ٥١٢٠ نافذة قامت بمسح النبضات المسجلة لقلب حصان واستخدام عينات تبلغ ٢٠٠ عينة في الثانية وتم اختيارها بدقة لتكون خالية من التشويش ثم تم تعليم وتدريب الشبكة لاستخلاص النبضات بعد إضافة الضوضاء إلى المدخل، ولقد فاقت هذه الشبكة عمل المرشحات الخطية المتكيفة (Adaptive Linear Filter) حيث أمكنها العمل إذا بلغت درجة التشويش ٥٠٪. وباستخدام نفس التقنية قامت شركة نستور (Nestor) بتطوير شبكة لتصنيف النبضات السليمة واكتشاف أية نبضات غير طبيعية.

(١٧-٦-٤) نظام النشر الإلكتروني للبيانات (Electronic Data Publishing System)

يتركب نظام النشر الإلكتروني شكل (١٧-١٨) من شبكة محلية نوفيل LAN-Novell

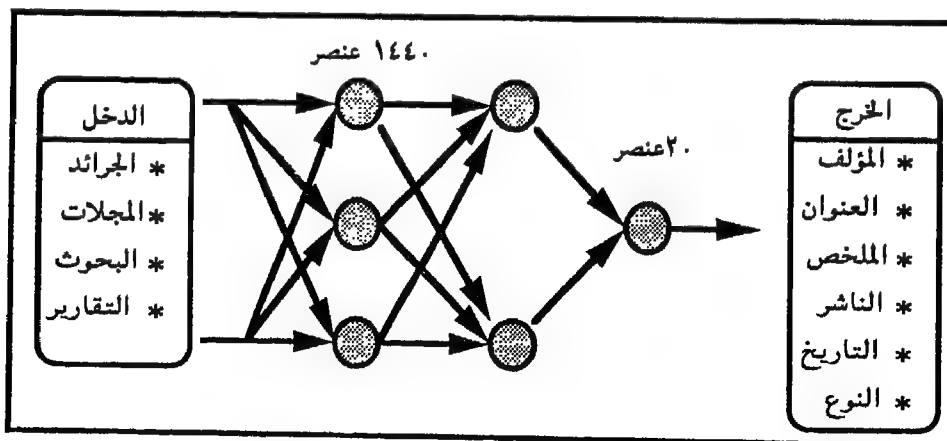
- ١ - مكونة من خادم للملفات و(٣٥) عقدة للاتصال بالحواسيب الأخرى ويتركب الجزء الأساسي من:
 - Scanner) والذي يقوم بإدخال صفحات من الجرائد والمجلات والبحوث والتقارير وماشابه ذلك إلى ذاكرة الحاسب، حيث يقوم بتعرف ضوئي للحروف (OCR) بالتعرف على الحروف والكلمات، ووضعها كدخل للشبكة العصبية والتي تقوم بالتعرف والتصنيف.
- ٢ - ولإزالة الخطأ في الكلمات المتشابهة عند القراءة من قبل الشبكة العصبية يستخدم أحد



شكل (١٧-١٨) شبكة محلية للنشر الالكتروني للبيانات

برامج تحويل الكلمات الى مقاطع صوتية (فونيم) مثل خوارزم (Soundex) الذي يوضح الفرق بين الكلمات المتشابهة مثل (Jonson, Johnson, Johnsen) ويؤدي الى الاقلال من عدد المتغيرات عند المدخل للشبكة العصبية.

٣ - وتتكون الشبكة العصبية شكل (١٧-١٩) من (١٤٤٠) عنصرا للدخال و (٢٠) عنصراً



شكل (١٧-١٩) الشبكة العصبية للنشر الالكتروني للبيانات

للمخرج وتستخدم لوحة الكترونية لتسريع العمليات الحسابية (Accelerator Board)، وتم تعليم الشبكة على كمية من البيانات تبلغ ١٠٠ ميجابايت، وبلغ عدد مرات التعليم والتدريب ١٠٠ مرة وبلغت نسبة التعرف الصحيح ٩٦٪. وتقوم الشبكة بتصنيف البيانات طبقا للنوع والمؤلف والناشر والعنوان والتخصص والتاريخ والملخص وغير ذلك ثم تقوم باستخدام برامج النشر المكتبي للكتابة ووضع المصنفات داخل نظام قاعدة البيانات من خلال الحاسبات الاخرى فى الشبكة المحلية.

ولقد وفرت هذه الشبكة فى أول شهرين من العمل ٢٠ الف دولار وقامت بمسح ٦٠٠ ميجابايت من المطبوعات الطبية التى تعادل ٢٠٠ الف صفحة .

الفصل الثامن عشر

شبكات كوهنن العصبية
ذاتية التنظيم

**Kohonen Self-Organising
Neural Networks**

(١٨-١) التعليم الذاتى (Unsupervised Learning)

كما رأينا سابقاً ان التعليم الموجه (Supervised Learning) من أهم السمات العامة للشبكات العصبية المتعددة الطبقات والتي تعتمد على تقنية الانتشار العكسى او الخلفى للتعليم، والتي بدورها تعتمد على الاستجابة الخارجية (External Response) لمصمم او معلم الشبكة الذى يحاول فى كل دورة تعليم ان يجعل الفرق بين الخرج الحقيقى والخرج المطلوب أقل ما يمكن، والتي أثبتت فى كثير من التطبيقات انها تعتمد على الاسلوب الانسانى للتعليم. ولقد وجد انه من المفيد فى كثير من التطبيقات ان تقوم الشبكة العصبية بتعليم نفسها ذاتياً على التصنيف والتعرف الصحيح وذلك يتطلب ان تتوفر فى الشبكة الشروط الآتية:

١- **الشرط الأول :** هو ان يجرى تعريف مجموعة البصمات التى تعرض عند مدخل الشبكة والتي تشترك فى مجموعة من الملامح والسمات المشتركة (Common Features) على انها تتبع نفس التصنيف لرتبة واحدة (Class Membership).

٢- **الشرط الثانى :** هو ان تكون الشبكة قادرة على تحديد للملامح المشتركة لمجموعة من البصمات التى تعرض عند مدخل الشبكة.

وبذلك يعرف التعليم الذاتى (Unsupervised Learning) بأنه التعديل الداخلى للاوزان من قبل الشبكة ذاتها لوضع النموذج الذى يوصف الملامح الممثلة للبصمات المدخلة عند إجراء التعليم، وتظهر أهمية التعليم الذاتى فى شبكات كوهن العصبية ذاتية التنظيم والتي تؤدى الى الحصول على خرائط تقوم بتنظيم نفسها ذاتياً ويطلق عليها خرائط كوهن ذاتية التنظيم (Kohonen's self-organising Maps).

(١٨-١-١) قروض التنظيم الذاتى (Self-organisation Concepts)

قام كوهن (Kohonen) استاذ المعلومات بجامعة هيلسنكى بالعمل المكثف فى بحوث الشبكات العصبية الاصطناعية لمدة كبيرة من الزمن وذلك فى مجال الذاكرة المترابطة (Associative Memory) والنمذجة الحسابية للأنشطة البيولوجية العصبية، ولقد تميز عمله باستحداث نماذج لعمليات التنظيم الذاتى (Self-organising) وطرق التعليم التكيفية بالمخ الانسانى. ومن المعروف إحتواء المخ الانسانى على مناطق مركزية العمل ومحددة المكان (Localised) ومسؤولة عن أداء بعض المهام مثل مهام التحدث والرؤية والتحكم فى الحركة، كما تتميز هذه المناطق بنشاط محلى (Local Activity) ذو تركيز عال.

كما أثبتت الأبحاث الحديثة ان هذه المناطق تحتوى على تركيبات فراغية يتأثر كل منها بمدى

معين من الاستجابة للإشارات المستقبلية من الاعضاء الحسية. مثال ذلك المناطق المسؤولة عن السمع والتي يمكنها تتبع الذبذبات الصوتية من خلال مقياس لوجارتمى حيث يتأثر أحد أطراف هذه التركيبات الفراغية بالذبذبات ذات التردد المنخفض ويتأثر الطرف الآخر بالذبذبات العالية التردد، وبذلك يمكن التعرف والاحساس بأى تردد بينهما.

ولقد تم اقتراح فكرة التنظيم الذاتى فى عام ١٩٧٣ من قبل العالم فون درمالسبرج والتي تبعها اقتراح نماذج تعمل بالتنظيم الذاتى والتي اعتمدت على بعض الأسس البيولوجية التى تشير الى وجود خاصية الاختيار لبعض النيورونات. مثال ذلك خاصية التأثر بشدة الضوء وكذلك حافة الاتزان فى المنطقة الخاصة بالرؤية، وكما تم ذكره من قبل ان عملية التعليم فى النظم البيولوجية تعتمد اعتمادا أساسيا على الضبط والتكيف الكيميائى لتعديل شدة تأثير المداخل العصبية عند الاتصال بالمشابك. ومن الاقوال المأثورة لمنسكى مايلى:

" اذا كان المخ الانسانى بسيط العمل والتركيب فانه يكون من البساطة ان نفهمه "

ولقد ذكر منسكى كذلك انه يوجد اكثر من ٤٠٠ معمارية مختلفة داخل المخ الانسانى، ولقد اعتمد كوهونن على ذلك عندما قام باستحداث واستنتاج خوارزم التعليم للتنظيم الذاتى. ولقد اعتمد الخوارزم المقترح على الفرض الاساسى الذى ينص على قيام المخ الانسانى بالتخطيط والتمثيل الفراغى الداخلى (Spatial Mapping) وذلك لوضع النماذج لتمثيل تراكيب البيانات المعقدة . ومن أهم خواص هذا الخوارزم مايلى:

١ - ضغط البيانات (Data Compression) من بيانات متعددة الابعاد (Multi-dimensional) الى بيانات ذات بعدين (Two-dimensional) حيث يجرى توزيع النيورونات فى طبقة واحدة ذات بعدين، بحيث يكون هناك تغذية عكسية جانبية، تكون لها القدرة على التعامل مع بيانات ذات أبعاد متعددة.

٢ - استخدام تقنية التوزيع الكمي للمتجهات (Vector Quantisation) وذلك لحزن البيانات فى الشبكة والتي تساعد على التمثيل الفراغى لبيانات متعددة الابعاد فى بعدين فقط.

(١٨-١-٢) شبكة كوهونن ذات البعدين (Two-dimensional Kohonen's Network)

يتركب الشكل العام لشبكة كوهونن والمبين فى شكل (١٨-١) من الاتى:

١ - طبقة واحدة ممتدة فى بعدين، ويلاحظ الاختلاف الواضح بين تركيب هذه الشبكة وبين الشبكة متعددة الطبقات من حيث عدم وجود الطبقات الثلاث (المدخل والمختفية والمخرج) بل يجرى توزيع العناصر الحسابية على مسطح فى بعدين.

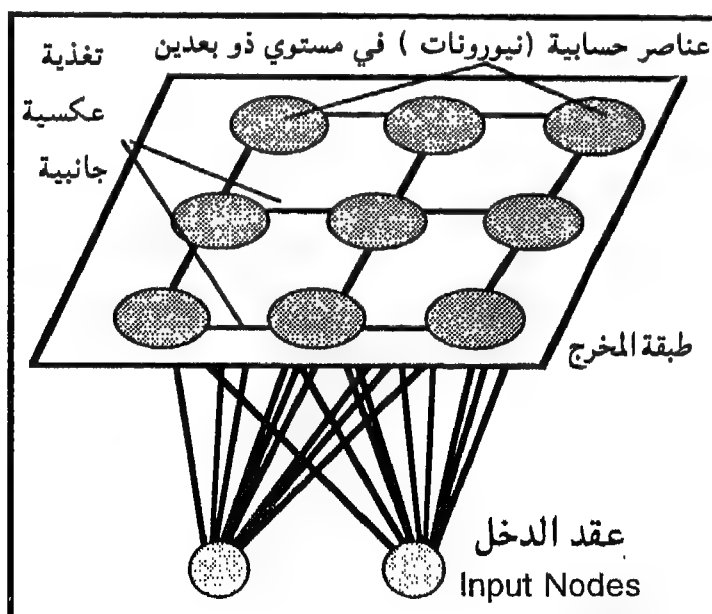
٢ - يجرى توصيل جميع المداخل من عناصر المدخل الى كل العناصر الحسابية فى المسطح ذو البعدين.

- ٣ - تقتصر التغذية العكسية على التوصيل الجانبي بين العناصر الحسابية الاقرب فقط .
- ٤ - لا يوجد للطبقة مخرج معين ولكن يتم اعتبار أى عنصر من العناصر فى المسطح كوحدة اخراج.
- ٥ - يطلق على هذا التركيب اسم خرائط الملامح لكوهن (Kohonen Feature Map) حيث انه يمكن خزن الملامح فى هذا التركيب الفراغى على شكل متجهات.

(١٨-٢) خوارزم كوهن للتعليم الذاتى (الغير موجه)

(Kohonen's Algorithm for Unsupervised Self-organised Learning)

يعمل خوارزم كوهن للتعليم الذاتى على تنظيم العناصر الحسابية فى المستوى المسطح للطبقة الثنائية الابعاد بحيث تصبح منطقة جوار (Neighbourhood) تقوم بالعمل كمصنف للملامح البيانات المدخلة (Feature Classifier) وذلك بمقارنة بيانات المتجه الذى يتم ادخاله والممثل لبصمة معينة مع متجه مخزون فى العناصر الحسابية لطبقة الاخراج وعند حدوث التطابق والمواءمة بين المتجهين فان العنصر الفائز سوف يقوم بنقل خواص المتجه المخزون من خلال خوارزم التعلم الى العناصر القريبة فى الجوار وبذلك تتحول طبقة الاخراج الى خريطة للملامح من خلال التعليم والتنظيم الذاتى.



شكل (١٨-١) - شبكة كوهن ذات البعدين
(Two-dimensional Kohonen's Network)

ويشتمل الخوارزم علي الخطوات التالية:

١ - تنشيط الشبكة وذلك بفرض ان قيم الاوزان هي $w_{ij}(t)$ وذلك في المدى $(0 \leq i \leq n-1)$ على ان تكون هذه الاوزان من العنصر i الى العنصر j وذلك عند زمن t شكل (١٨-٢) وذلك كمايلي:

(أ) يتم فرض قيم عشوائية صغيرة لعدد n من المداخل لكل عنصر حسابي.

(ب) يتم تعيين واختيار دائرة الجوار حول عنصر j ليضم مساحة كبيرة ويرمز لهذا الجوار

الابتدائي بالرمز $N_j(0)$

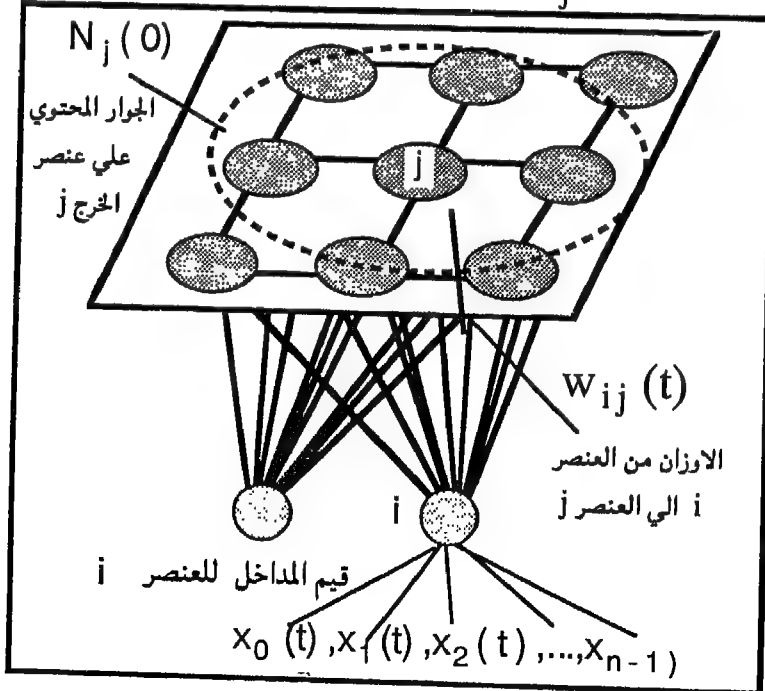
٢ - يتم تعيين قيم المداخل في الصورة : $x_0(t), x_1(t), x_2(t), \dots, x_{n-1}(t)$ حيث $x_j(t)$ هي قية المدخل لعنصر المدخل i عند الزمن t .

٣ - يتم حساب المسافة d_j بين المدخل وكل عنصر من عناصر المخرج j كما في شكل (١٨-٣) كالآتي:

$$d_j = \sum_{i=0}^{n-1} (x_i(t) - w_{ij}(t))^2$$

٤ - يتم تحديد اقل مسافة وتعيين عنصر المخرج الذي يقع على هذه المسافة وليكن j^* .

٥ - يتم ضبط الاوزان للعنصر j^* وكذلك لجميع العناصر في الجوار الذي يحتوي على هذا العنصر



شكل (١٨-٢) - تنشيط الاوزان واختيار منطقة الجوار

الفائز، والذي يرمز له بالرمز $N_j^*(t)$ وتصبح الاوزان الجديدة كمايلي:

$$w_{ij}(t+1) = w_{ij}(t) + \eta(t)(x_i(t) - w_{ij}(t))$$

وذلك للعنصر j فى الجوار $N_j^*(t)$ ، مع العلم بان $(0 \leq i \leq n-1)$

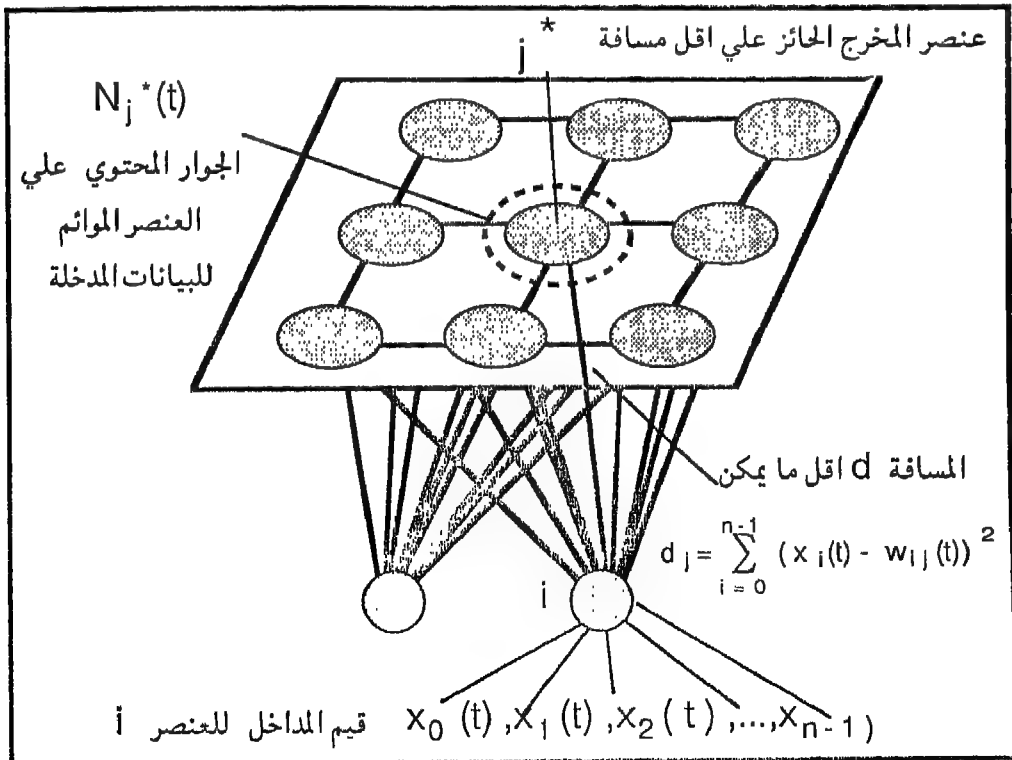
حيث يمثل المعامل $\eta(t)$ معاملا للمكسب (Gain) والتي تنحصر قيمته بين الصفر والواحد الصحيح والذي تقل قيمته مع كل دورة ضبط للاوزان.

٦ - يلاحظ ان دائرة الجوار تقل فى المساحة لتشمل اقل عدد من العناصر التى تتشابه وتتوائم مع بيانات وملامح مدخل معين وبذلك أمكن انشاء منطقة او جوار نشط متشابه.

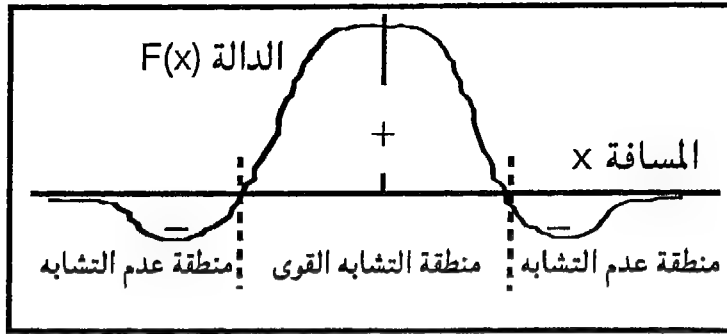
٧ - يجرى تكرار الخطوات من الخطوة (٢) لادخال بيانات اخرى.

(١٨-٢-١) التوصيل التبادلى الجانبي (Lateral Interconnection)

لقد بنى كوهنن معظم أفكاره على ما يحدث فى المخ الانسانى من وجود مناطق محددة ذو نشاط خاص كما أوضحنا قبل ذلك، ولنا ان نتساءل كيف تؤثر المسافة الجانبية او التوصيل الجانبي



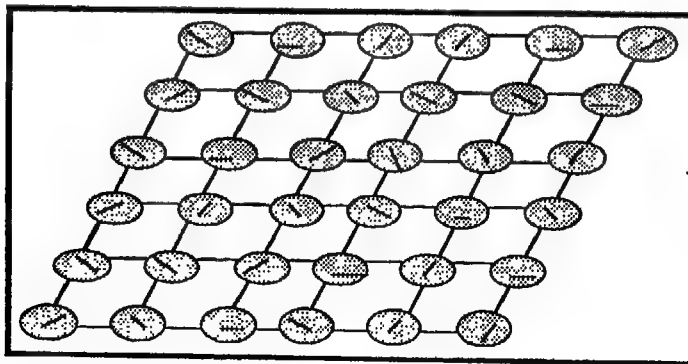
شكل (١٨-٣) - تحديد العنصر الموائم (الفائز) ومنطقة الجوار المثلى



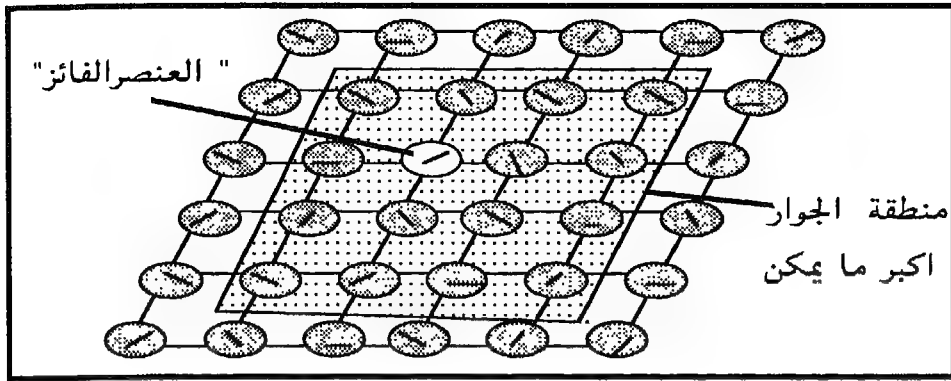
شكل (١٨-٤) التوصيل الجانبي ودالة القبة المكسيكية

على قدرة التوصيل للنيرون وتقول احد الفروض انه يمكن تصوير ووصف دائرة النشاط في المخ الانساني بدالة القبة المكسيكية ($F(x)$) والمبينة في شكل (١٨-٤) حيث تتكون منطقة النشاط من الخلايا القريبة من النيرون النشط وذلك نظرا لقوة الترابط ثم يتلاشى النشاط مرة واحدة بعد ذلك حيث يؤدي الترابط الضعيف الى خلق مناطق عدم نشاط على مسافة ليست بعيدة من بؤرة النشاط الاصلى ويساعد هذا الفرض على تفهم كوهن لعمل منطقة الجوار عند تنفيذه لخوارزم شبكته ذات التعلم والتنظيم الذاتى.

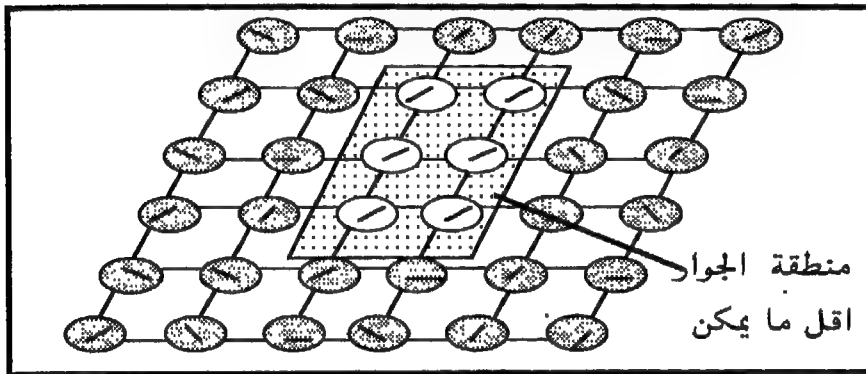
من الواضح ان خوارزم هذه الشبكة يتميز بالسهولة نظرا لسهولة قاعدة التعليم حيث لايتطلب اجراء عملية التفاضل كما هو معروف فى خوارزم الشبكة المتعددة الطبقات، بل يبدأ تحديد اوزان عشوائية وذات قيم صغيرة لجميع الوصلات بين المداخل والعناصر الحسابية وبذلك يكون هناك متجه وحيد لكل عنصر يمكن خزنة داخل هذا العنصر كما هو مبين فى شكل (١٨-٥) أ ، وتبدأ عملية التعليم بمقارنة جزء من بيانات البصمة المدخلة ومقارنتها بالمتجهات



شكل (١٨-٥) أ تحديد الازان عشوائيا وخزن المتجهات فى العناصر



شكل (١٨-٥) ب تحديد " العنصر الفائز "



شكل (١٨-٥) ج العناصر المجاورة ذات اوزان متشابهة محاطة بمنطقة
عدم تشابه مثل القبة المكسيكية

المخزونة داخل هذه العناصر كل على حدة الى ان يتم الوصول الى " العنصر الفائز " وهو العنصر الذى يحتوى على متجه مشابه او قريب الشبه ببيانات متجه المدخل، ثم يقوم العنصر الفائز بعملية ضبط لقيمة الاوزان المخزونة به مع بيانات المدخل حتى يصبح مثله تماما، وبذلك يعطى هذا العنصر استجابة كاملة للتعرف على هذه البيانات عند عرضها مرة ثانية كما هو موضح فى شكل (١٨-٥) - ب.

ثم يجرى تعديل الاوزان للعناصر المجاورة لهذا العنصر لتأخذ الاتجاه للعنصر الفائز تقريبا، والذى يعنى ان العناصر المجاورة سوف تقوم بعمل المتوسط للمتجه الفائز، وبذلك يمكن للشبكة ان تتعرف على حالات قريبة الشبه من الحالات المدخلة حتى اذا لم تكن تعلمت عليها قبل ذلك شكل (١٨-٥) - ج.

(١٨-٣) التعليم باستخدام التوزيع الكمي للمتجهات

(Learning by Vector Quantisation (LVQ))

لقد قام كوهن بتطوير طريقة التعليم باستخدام التوزيع الكمي للمتجهات والتي تعتبر طريقة للضبط الدقيق لتعليم خريطة الملامح للوصول الى التعرف الامثل، وقد تنشأ بعض الظروف التي تتطلب اضافة متجه معين يساعد من كفاءة التعرف لاحد المناطق المعينة فى جوار معين، ولتنفيذ ذلك يمكن اتباع الخطوات الآتية :

- ١ - يتم اختيار متجه معين معروف التصنيف وليكن (x) .
 - ٢ - يجري تعريض هذا المتجه عند مدخل الشبكة العصبية وذلك لاكتشاف دقة التعرف والتصنيف .
 - ٣ - لنفرض ان العنصر الحسابي الفائز عند اجراء المقارنة والذي يتواءم مع المتجه المخزون به مع المتجه المعروض عند المدخل هو العنصر (n_w) .
 - ٤ - يجرى تعديل الوزن بالنسبة للعنصر الفائز وذلك طبقاً للقاعدة الآتية :
- أ - يكون التعرف والتصنيف صحيحاً اذا توفرت القاعدة :

$$n_w(t+1) = n_w(t) + \eta(t) [x(t) - n_w(t)]$$

حيث يقوم الحد $\eta(t)$ بالتحكم فى معدل الضبط كما يحدث تماماً فى احدى دورات التعليم ويمكن الاستعانة بهذه الطريقة لاضافة متجهات عند تطوير نظم التعرف على التحدث مثلاً وذلك لضبط الأصوات الجديدة لمستخدمين جدد .

ب- يكون التعرف غير صحيحاً اذا توفرت القاعدة:

$$n_w(t+1) = n_w(t) - \eta(t) [x(t) - n_w(t)]$$

(١٨-٤) بعض تطبيقات شبكات التعليم والتنظيم الذاتى

(١٨-٤-١) النشر المكتبى باستخدام الصوت

(Phonetic Word- processing)

قام كوهن بعمل تجربة تطبيقية ناجحة باستخدام شبكته ذات التعليم والتنظيم الذاتى فى مجال التعرف على الحديث والكلام والذي يعتبر من المجالات المعقدة نسبياً نظراً لكونها معالجة للتعرف والتصنيف فى الزمن الحقيق (Real Time) وذلك لاثبات صحة وكفاءة كل من خوارزم التعليم الذاتى باستخدام التوزيع الكمي للمتجهات (LQV) وتقنية انشاء خرائط التعرف (Feature Map Techniques) .

ولقد حالفه الحظ في اختيار هذا التطبيق الذي يعتبر تحدياً حقيقياً حيث يتطلب تطوير واستحداث خريطة تكون قادرة على التعرف على الملامح الصوتية التي تحتوى على جميع الفونومات الصوتية التي تختلف عن بعضها البعض اختلافاً كبيراً.

ومن المعروف ان التعرف على الحديث بالنسبة للانسان العادى يحتاج الى الكثير من المعارف مثل معرفة سياق الكلام (Context) والاستدلال (Inference) والاعراب (Parsing) والتوقع لما يمكن ان يقوله المتحدث (Extrapolation) وقواعد النحو (Syntactic Rules) كما يتفوق الانسان في التعرف الصوتى حتى في وجود تشويش وضوضاء عالية. ومن ناحية التحليل الالكترونى للاشارات الصوتية والفونومات فان التردد والشدة الصوتية تختلف من متكلم الى آخر وحتى للمتكلم نفسه فقد تختلف هذه الخواص وقد تتداخل الخواص الترددية للفونومات. ولقد امكن كوهن من التغلب على كثير من العقبات لبناء الآلة الكاتبة التي تعمل بالصوت (Phonetic Typewriter) وذلك من خلال الاملاء، وبين شكل (١٨-٦) تفاصيل هذا النظام والذي يعتبر نظام خبرة يعتمد على القواعد حيث تقوم فيه الشبكة العصبية بالعمل الاساسى للتصنيف والتعرف. وقد اعتمد هذا النظام كذلك على التحليل الالكترونى للموجات الصوتية باستخدام التقنية الرقمية ويتركب النظام من:

أ- المعالجة ابتدائية (Pre processing) والتي تشمل الخطوات الآتية :

١- الترشيح : يتم ترشيح الاشارات المستقبلية من الميكروفون باستخدام مرشح منخفض

التردد (Low - pass Filter) 5.3 KHz

٢- التحويل من تماثلى الى رقمى : وذلك باستخدام محول ١٢ بت التماثلى الى

رقمى (A/D 12 - BIT) والذي يقوم بتقطيع الاشارة الى عينات تبلغ

معدلها ١٣.٣٠ عينة في الثانية (Sampling rat 13.03KHz) .

٣-تحويل فورير السريعة ((Fast Fourier Transform (FFT) : وباستخدام

هذه التحويرة يمكن حساب عدد ٢٥٦ نقطة من البيانات الرقمية الخارجة من المرشح

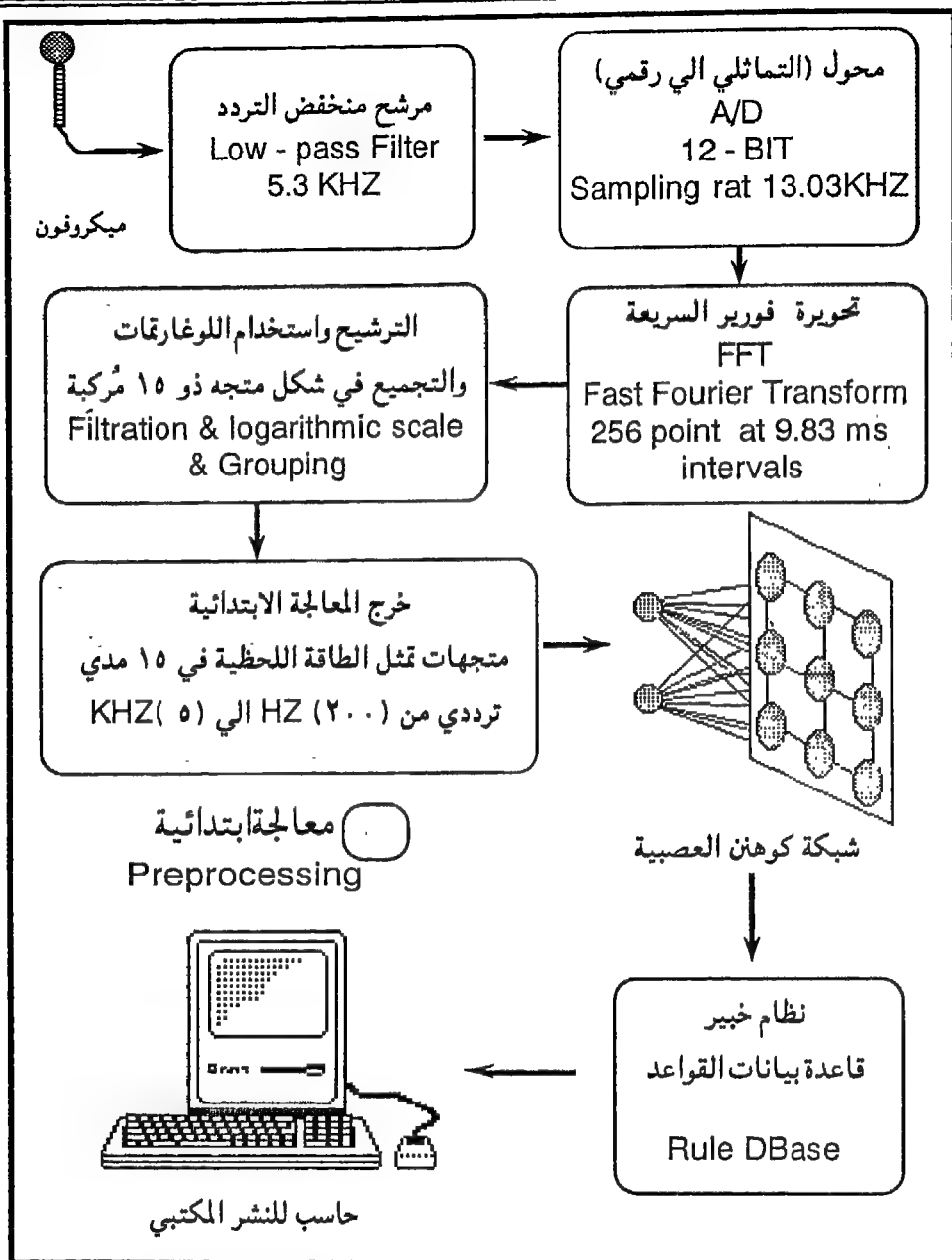
وذلك على مسافات زمنية تبلغ ٩,٨٣ م. ثانية (256 point at 9.83 ms intervals)

٤ - الترشيح واستخدام اللوغاريتمات والتجميع (Filtration & logarithm- والتجميع

(mic scale & Grouping) وتشمل هذه الخطوة ترشيح الخرج من تحويل فورير

السريعة وتحويلها تحت مقياس لوغاريتمى ووضعها فى شكل متجهات تبلغ عدد مركبات

كل متجه ١٥ مركبة تمثل الطاقة اللحظية فى عدد ١٥ مدى ترددى من ٢٠٠ (HZ)



شكل (١٨-٦) النشر المكتبي باستخدام شبكة كوهنن ذاتية التعليم

الى ٥ (KHz)، كما تم اضافة المركبة السادسة عشرة لكل متجه والتي تشمل معلومات عن الاشارة مثل الجذر التربيعي لمتوسط مجموع مربع السعات للاشارة.

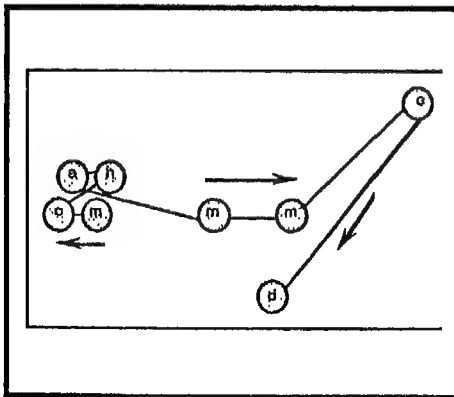
ب - التعليم الذاتى للشبكة :

كما رأينا فى المعالجة الابتدائية ان الصوت قد تم توزيعه كمياً ونحويلة الى متجهات تحمل الملامح ذات (١٦) مركبة حيث يمكن اعتبار هذا المتجه كجزء متناهى الصغر من شكل الحديث والذى يحمل صفاته الترددية. ولقد تم تعليم الشبكة للتعرف على هذه المتجهات والتي توصف الفونومات، وتم استخدام عدد ٥٠ عينة لكل فونم لكي تتعلم الشبكة، ولقد حصل كوهن على مايسمى بالخريطة الصوتية والمبينة فى شكل (١٨-٧).

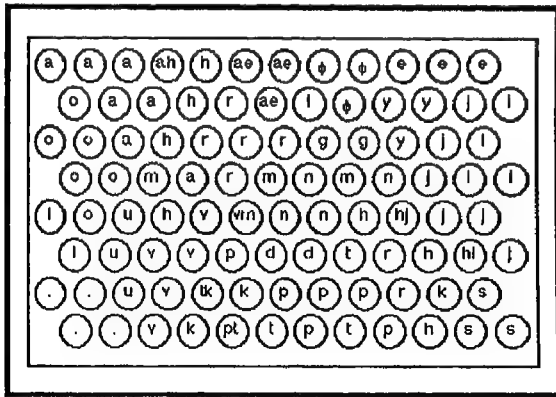
وبين شكل (١٨-٨) تمثيل لاستخدام الخريطة الصوتية لبيان كيف يمكن التعرف على كلمة محمد باللغة الانجليزية، ولقد اقترح كوهن انه يمكن استخدام هذه الطريقة لتعليم المصابين بداء الصمم.

ج - المعالجة النهائية (Post Processing)

تعتبر عملية الترجمة من التعرف الصوتى للمتحدث الى الكتابة على الحاسب هى طور المعالجة النهائية، وذلك باستخدام نظام خبير يعتمد على القواعد، والذى يقوم باعطاء النحر الصحيح المقابل للفونومات المستقبلية من المتحدث بعد التعرف الصوتى عليها. ولقد احتوى هذا النظام على عدد من القواعد يتراوح بين ١٥ الى ٢٠ الف قاعدة، والتي اعتمدت أساسا على حساسية السياق (Context Sensitivity) للفونيمات. ولقد اعتمد كوهن على استخدام قاعدة بيانات تم استحداثها من الأمثلة لبيانات الحديث الحى المعتمد على النطق والكتابة للفونيمات الصحيحة. ولقد بلغت سرعة الكتابة على الحاسب ١٠٠ كلمة فى ١٠ دقائق، وذلك باستخدام حاسب شخصى والاستعانة بمعالج رقمى للإشارات (TMS32010) لتحليل الصوت.



شكل (١٨-٨) التعرف على كلمة محمد



شكل (١٨-٧) الخريطة الصوتية لكوهن

الفصل التاسع عشر

شبكات هوبفيلد العصبية
ذات الاتصال الكامل

**Hopfield Fully-Connected
Neural Networks**

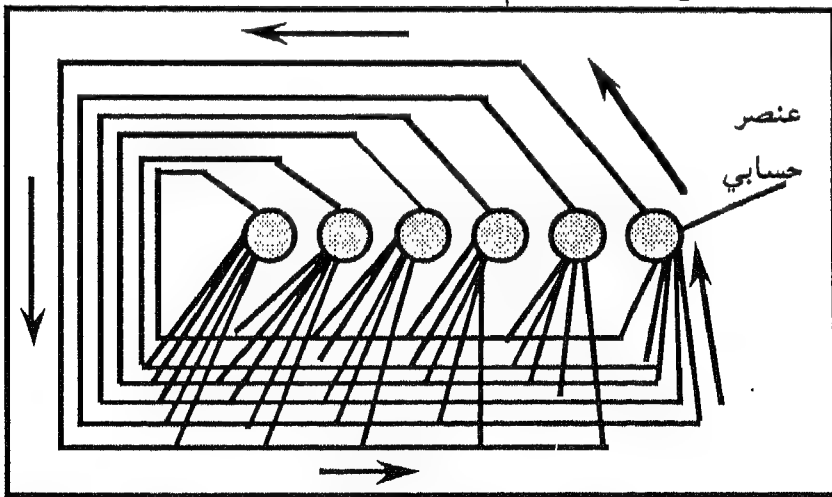
(١٩-١) شبكات هوبفيلد (Hopfield Networks)

قام هوبفيلد منذ عام ١٩٨٠ بدراسة مكثفة للشبكات العصبية الاصطناعية ذاتية الترابط (Autoassociative Nets) والتي اعتمدت أساسا على ربط بارامترات الشبكة بدالة الطاقة (Energy Function) والنظم الفيزيائية (Physical Systems) الاخرى.

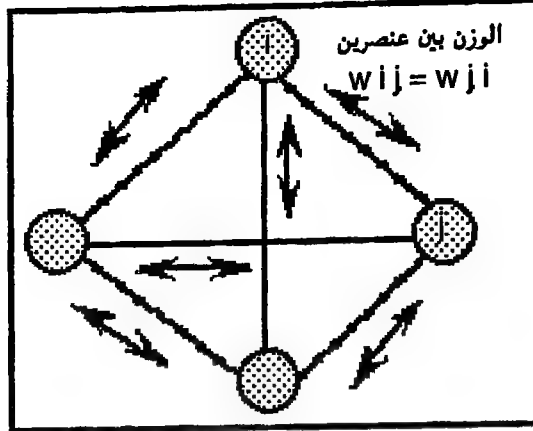
(١٩-١-١) تركيب (معمارية) الشبكة (Network Structure)

تتركب الشبكة من عدد من العناصر الحسابية بحيث يتصل كل عنصر بجميع العناصر الاخرى (Fully Connected Network) ويبين شكل (١٩-١أ) طريق توضيح الاتصال الكامل بين العناصر المختلفة، كما يوضح شكل (١٩-١ب) طريقة اخرى لتوضيح الاتصال الكامل بين هذه العناصر. ومن أهم خصائص الشبكة مايلي:

- ١ - تتماثل وتتساوى الاوزان بين كل عنصرين اى ان $(w_{ij} = w_{ji})$ ولذلك يطلق على هذه الشبكة " الشبكة ذات الاوزان المتماثلة " (Symmetrically-Weighted Network).
 - ٢ - يجري تطبيق قواعد النيرون المفرد على عناصر هذه الشبكة من حيث المجموع الموزن والدالة الحدية العتبية الشكل (Threshold Step - Function).
 - ٣ - يقوم كل عنصر على حدة بحساب المجموع الموزن للمداخل وطرح القيمة الحدية من ذلك المجموع وتطبيق الدالة العتبية على الناتج وذلك لتحديد قيمة الخرج.
 - ٤ - تقبل الشبكة دخلا فى أحد الشكلين:
- أ - الشكل الثنائى للأرقام ((0-1) Binary).
- ب - الشكل ثنائى القطب للأرقام ((Bipolar)) (+1, -1).



(١٩-١أ) شبكة هوبفيلد العصبية ذات الاتصال الكامل



(١٩-١) توضيح الاتصال الكامل لشبكة هوبيلد العصبية

مع ملاحظة ان الشكل الثنائي القطب يسهل عملية اجراء الحساب.
٥ - لا يتم تخصيص عنصر للمدخل ولا عنصر للمخرج كما هو واضح فى الشكل (١٩-١) حيث تتشابه العناصر مع بعضها البعض حيث تعتبر هذه الظاهرة من سمات شبكة هوبيلد الكاملة الاتصال. من الواضح ان تركيب ومعمارية الشبكة يختلف اختلافا اساسيا عن التركيب والمعماريات للشبكات الاخرى، وبذلك يكون من المتوقع ان تكون طريقة عمل هذه الشبكة مختلفاً كذلك.

ويمكن تلخيص خطوات العمل للشبكة فى الخطوات التالية :

- ١ - يجرى ادخال قيم المداخل التى تتراوح بين (+1, -1) لجميع العناصر فى نفس الوقت .
- ٢ - تترك الشبكة لتقوم بعمل دورة حسابية (Cycle) حيث تمر بعدد من الحالات المتتالية والتى تعتبر أوضاع غير مستقرة الى ان تصل الى وضع مستقر (Steady State) يمثل الحل المناسب حيث يطلق على هذا الوضع ان الشبكة قد وصلت الى حالة التجمّع (Convergence) وذلك لثبات القيم الحسابية للعناصر.
- ٣ - تعتبر القيم المخزونة بجميع العناصر هى قيم المخرج المطلوب.

(١٩-٢) سلوك الشبكة ذات الاتصال الكامل

(Behaviour Of Fully Connected Network).

يمكن وصف السلوك العام للشبكة فى البنود التالية:

- ١ - تؤثر خاصية الاتصال الكامل للشبكة (Fully Connected Network) على سلوك الشبكة، حيث تقوم القيمة الموجودة بكل عنصر حسابى من التأثير على جميع القيم الحسابية للعناصر الاخرى.
- ٢ - تبدأ الحالة الابتدائية الغير مستقرة للشبكة بوجود مجموعة من القيم المختلفة بالعناصر

الحسابية التى تحاول ان تؤثر علي بعضها البعض، وبذلك يكون كل عنصر واقع تحت تأثير قيم تعمل على تحويله الى الحالة النشطة (on) مثلاً بينما تعمل قيم اخرى على تحويله الى حالة الخمود (off).

٤ - تستمر الشبكة فى العمل المتتالى بين الرفع والخفض الى ان تصل الى حالة من الثبات، حيث تثبت القيم لكل عنصر، وذلك يؤكد انه قد تم الوصول الى توازن بين عدد العناصر التى تحاول ان تجعل هذا العنصر نشطاً وبين العدد الآخر الذى يحاول ان يجعله خامداً.

٥ - يجب ملاحظة ان طريقة تعليم شبكة هوبفيلد تختلف جوهرياً عن طريقة تعليم الشبكة المتعددة الطبقات وذلك بناءً على نوعية التركيب، حيث تعمل جميع العناصر فى اول الامر كعناصر للدخال ثم تصبح عناصر للخارج ثم تستمر فى العمل بين الادخال والاخراج حتى يتم الحصول على الخرج المطلوب.

(٣-١٩) خوارزم هوبفيلد للشبكات ذاتية الترابط

(Hopfield 's Algorithm for Autoassociative Network)

يمكن تلخيص خطوات خوارزم هوبفيلد للتعليم للشبكات ذاتية الترابط فى الخطوات الآتية:

١ - نفرض ان هناك عدد (M) من البصمات التى تبدأ من الرتبة (s=0) الى الرتبة (s=M-1)، وان حد الاثارة لجميع العناصر هو الصفر.

٢ - نفرض (x^s_i) هو الجزء من البصمة فى الرتبة (s) والتى تاخذ احدى القيم (1-) او (1+)

٣ - وعلى ذلك يمكن حساب الاوزان للوصلات (Connection Weights) فى الصورة الآتية:

$$W_{ij} = \begin{cases} \sum_{s=0}^{M-1} x_i^s x_j^s & i \neq j \\ 0 & i = j, 0 \leq i, j \leq M-1 \end{cases}$$

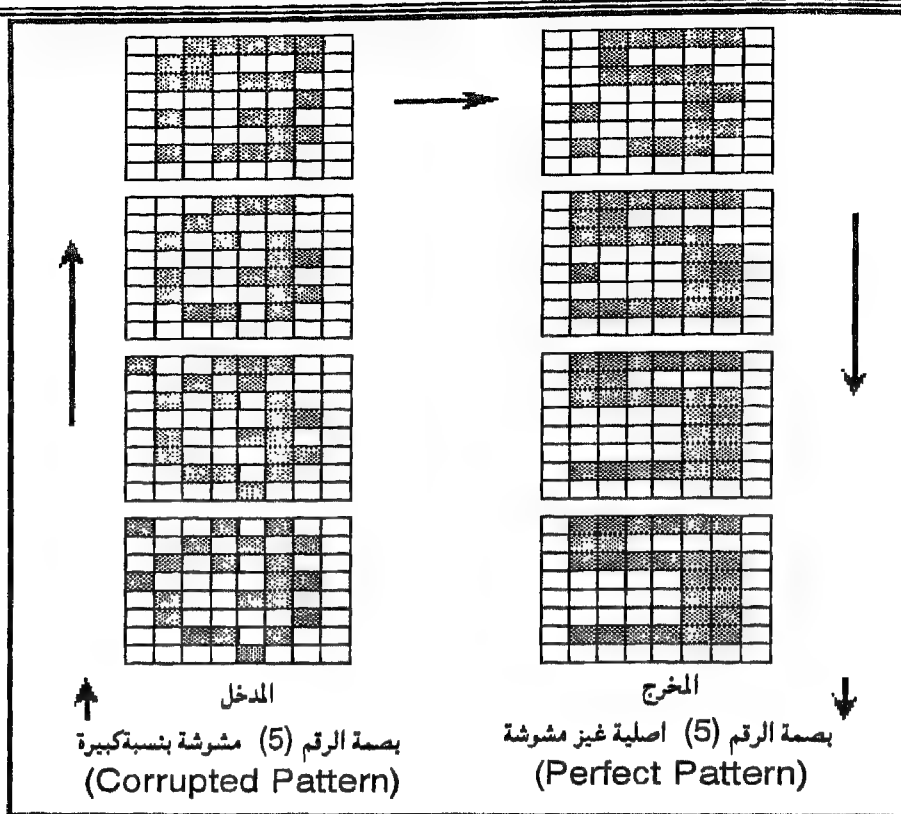
حيث W_{ij} هو وزن الوصلة بين العنصرين (i) و (j).

٤- إبدأ بالتنشيط لبصمة غير معرفة فى الشكل : $\mu_i(0) = x_i$

حيث $0 \leq i \leq N-1$ وكذلك $\mu_i(0)$ و $\mu_i(t)$ هى قيم المخرج من العنصر (i) عند الزمن (0) والزمن (t).

٥- يتم ضبط قيم الخرج عدة مرات الي ان يثبت عند قيمة واحدة وذلك باستخدام العلاقة الآتية:

$$\mu_i(t+1) = f_h \left\{ \sum_{j=0}^{N-1} w_{ij} \mu_j(t) \right\} \quad 0 \leq j \leq N-1$$



(٢-١٩) خاصية الترابط الذاتي لشبكات هوبفيلد

(Autoassociative Property for Hopfield 's Network)

حيث (f_n) تعبر عن الدالة الحدية العتبية الشكل (Threshold Step - Function).

ومن الملاحظ ان مرحلة التعليم هي المرحلة التي يتم فيها ضبط الاوزان بين العناصر على بصمة معروفة الى ان تثبت قيم المخرج عند اجراء الضبط مرة اخرى، وتعتبر مرحلة التعرف هي المرحلة التي يتم فيها عرض بصمة غير معروفة، حيث تقوم الشبكة بمحاولة ضبط نفسها عدة مرات الى ان تصل الى حالة التعرف وذلك عند ثبات القيم عند المخرج حيث يقال ان الشبكة قد تجمعت وصلت الى الحل.

ويمكن تفسير وصف الشبكة بانها ذاتية الترابط (Autoassociative) كما يلي:

ان الشبكة قادرة على اعطاء بصمة اصلية سليمة (Perfect Pattern) غير مشوشة عند المخرج اذا تم ادخال نفس البصمة مشوشة (Corrupted Pattern) عند المدخل كما هو واضح في الشكل (٢-١٩) حيث يبين ان البصمة المدخلة للرقم (5) والمشوشة بشكل كبير يصعب على الانسان تصويره من أول وهلة، الا ان الشبكة سوف تمر بعدة مرات من الضبط حتى تصل الى

البصمة الغير مشوشة والمخزونة بداخلها قبل ذلك، ولذا فان عمل الشبكة يماثل عمل الذاكرة المعنونة (Content - Addressable Memory) حيث يمكن للانسان الحصول على محتوياتها من خلال العنوان الخاص بها.

(١٩-٤) مستويات الطاقة لشبكات هوفيلد

(Energy Landscape for Hopfield's Network)

يمكن تفهم طريقة عمل الشبكات العصبية ذاتية الترابط وذلك من خلال الإستعانة بما يسمى بمستويات الطاقة (Energy Landscape) والذي يعطى تصور لسلوك الشبكة من خلال تمثيل جميع الحالات الممكنة للقيم التى يمكن لدالة الطاقة ان تأخذها، وبذلك يمكن الحصول على مسطح يمثل مستويات الطاقة حيث تمثل القمم (Hills) الحالات البعيدة عن اتزان الشبكة وتمثل القيعان (Basins) الحالات القريبة من الاتزان للشبكة ويكون هناك قاع واحد (Bottom) تكون فيه الطاقة اقل مايمكن والذي يمثل حالة الاتزان للشبكة (Stable State) والذي يعرف ببئر الاتزان. ولنا ان نتصور مستويات الطاقة فى شكل قطعة من الارض التى تحتوى على قمم وأخاديد، ويجرى تمثيل دالة الطاقة للشبكة بكرة صغيرة وان هذه الكرة تبحث عن الاستقرار فى أعماق أخدود محتمل والذي يمثل الحل وان عملية التعليم هى محاولة وضع الكرة فى هذا الاخدود وذلك بالسير فى اتجاهه وبالعمل على تقليل قيمة دالة الخطأ الى أقل مايمكن.

ويمكن تفهم هذا الوضع بالتفصيل كمايلى:

١ - كما ورد سابقا فى الشبكات العصبية المتعددة الطبقات من حساب دالة الخطا فى الشكل:

$$E = \frac{1}{2} \sum (t_{pj} - o_j)^2$$

والتي تعتمد على المعرفة التامة بالخرج المستهدف او المطلوب وكذلك الخرج الحقيقى فى كل دورة تعليم والتي تناسب تركيب ومعمارية هذه الشبكات .

٢ - قد يختلف الوضع بالنسبة للشبكات ذاتية الترابط لهوفيلد حيث تقوم الشبكة باتخاذ الخطوات المناسبة فى اتجاه الوصول الى الحل ونظرا لعدم امكانية تطبيق دالة الطاقة المذكورة اعلاه لعدم مناسبتها لهذا النوع من الشبكات فانه يجرى تعريف دالة الخطأ لهذه الشبكات والتي لايد ان تحتوى وتأخذ فى الاعتبار مايلى:

أ - ملامح دالة الطاقة للنيرون والتي لا بد ان تكون ذات قيم كبيرة عندما يكون الخطأ كبيرا وذات قيم صغيرة عندما يكون الخطأ صغيرا.

ب- قيم الاوزان المثلة للبصمة المدخلة.

٣ - يمكن فرض دالة الطاقة التى تاخذ ماتم ذكره من اعتبارات لتناسب شبكة هوبفيلد فى الشكل

$$E = -\frac{1}{2} \sum_i \sum_j w_{ij} x_i x_j + \sum_i x_i T_i$$

الآتى:
حيث:

(w_{ij}) = الاوزان بين العنصر (i) والعنصر (j).

(x_i) = قيمة الخرج من العنصر (i).

(T_i) = القيمة الحدية للعنصر (i).

٤ - طبقا لاستخدام التغذية العكسية حيث تصيح قيم الخرج فى اى لحظة هى قيم الدخل فى الدورة التالية والذي يودى الى ان تقوم الاوزان والمدخلات بالتمثيل الصحيح للبصمة المدخلة.

٥ - يجرى استخدام طريقة انخفاض الميل (Gradient Descent) لدالة الخطا حتى نصل الى القاع المثل لحالة اتزان الشبكة (اقل طاقة)والذى يمثل الحل .

(١٩-٤-١) ميكانيكية التخزن للبصمات (Patterns Storing Mechanism)

مما تم عرضه يمكننا القول بانه لابد من العمل على تخزين البصمات داخل الشبكة، ولكى يتم ذلك فاننا نقوم باتباع الخطوات السابقة حتى يتم الوصول الى اقل طاقة ممكنة لبصمة معينة والى يمكن تمثيلها بنقطة تحتل احد الآبار فى مسطح مستويات الطاقة والتي يجرى تخزينها فى الشبكة بحيث لاتؤثر على القيم للبصمات المخزونة من قبل والخاصة ببصمات اخرى.

وللوصول الى ذلك نفرض مايلى:

١- ان هناك بصمة معينة ولتكن (S) وان عناصر الادخال لهذه البصمة هى:

$$(x_0, x_1, x_2, x_3, \dots, x_{n-1})$$

٢- نفرض ان دالة الطاقة والتي يراد ان يتم تعديلها لتكون اقل ما يمكن فى الشكل :

$$E = -\frac{1}{2} \sum_i \sum_j w_{ij} x_i x_j + \sum_i x_i T_i$$

ولكى تصيح كذلك فلا بد للجزء الثانى الموجب من المعادلة ان يختفى ويتم ذلك باختيار

القيمة الحدية (T_i) لجميع العناصر الحسابية بالقيمة صفر ويبقى الجزء الاول فقط كمايلى:

$$E = -\frac{1}{2} \sum_i \sum_j w_{ij} x_i x_j + \sum_i x_i T_i$$

٣ - نفرض انه يمكن كتابة عنصر الادخال للبصمة (S) فى الشكل (x_i^S) والذي يمكن ان ياخذ

القيم (+1) او (-1)، وكذلك الوزن (w_{ij}) كما تم تعريفه سابقا على انه الوزن بين العنصرين (i) و (j) والذي يشتمل على معلومات عن جميع البصمات المراد تعلمها من قبل الشبكة.

٤ - بهذا الاسلوب يمكن فصل مصفوفة الاوزان الى مصفوفتين تمثل المصفوفة الاولى الاوزان (w_{ij}) من جميع البصمات الاخرى ماعدا البصمة (s) وتمثل المصفوفة الثانية الاوزان (x_i^s)

الخاصة بالبصمة (s) فقط وتصبح الصورة كمايلي:

$$E = -\frac{1}{2} \sum_i \sum_{j \neq i} w'_{ij} x_i x_j - \frac{1}{2} \sum_i \sum_{j \neq i} w^s_{ij} x_i^s x_j^s$$

$$= E_{\text{ALL EXCEPT } s} + E_s$$

$$= E_{\text{Noise}} + E_{\text{Signal}}$$

والتي يمكن تصورها على انها الطاقة الناتجة من الاشارة، وفي هذه الحالة هي البصمة (s) مضافا اليها طاقة الضوضاء.

٥ - وتعتبر ميكانيكية خزن البصمة (s) في الحقيقة ماهي الامحولة لتقليل قيمة الطاقة الناتجة عن البصمة الى اقل مايمكن اى الوصول الي اقل قيمة للمعادلة الاتية :

$$E_s = -\frac{1}{2} \sum_i \sum_{j \neq i} w^s_{ij} x_i^s x_j^s$$

ونظرا لوجود الاشارة السالبة والتي تعني ان تكون القيمة الاتية اكبر مايمكن:

$$\sum_i \sum_{j \neq i} w^s_{ij} x_i^s x_j^s$$

٦ - ونظرا لانحصار قيم الادخال للمتغير (x_i) بين (+1) و (-1) فاننا نستخدم القيمة التربيعية له في الشكل (x_i^2) والتي تكون موجبة دائما ويتم ذلك بالتعويض الآتي:

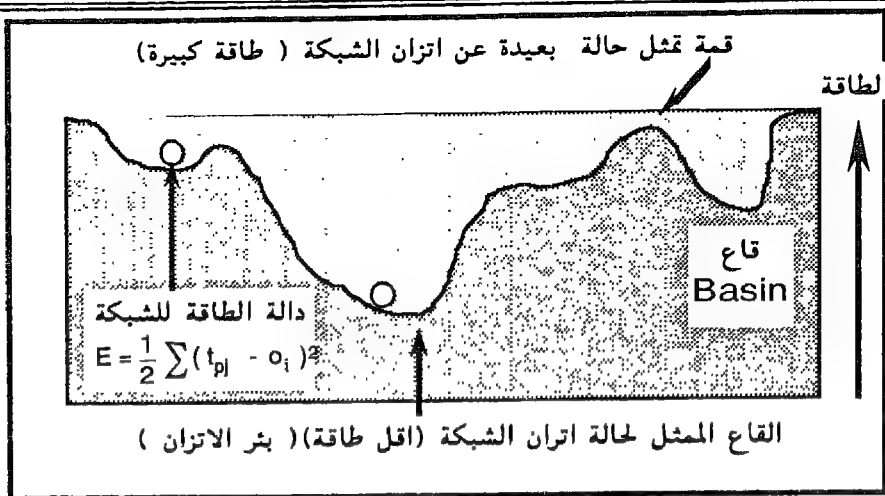
$$\sum_i \sum_{j \neq i} w^s_{ij} x_i^s x_j^s = \sum_i \sum_{j \neq i} x_i^2 x_j^2$$

مع ملاحظة انه يجب كتابة الوزن (w^s_{ij}) في الشكل الآتي:

$$w^s_{ij} = x_i x_j$$

ولايجاد الاوزان لجميع البصمات فانه يمكن اجراء الجمع على كل البصمات كمايلي:

$$w_{ij} = \sum_s w^s_{ij} = \sum_s x_i^s x_j^s$$



(٣-١٩) مستويات الطاقة لشبكات هوبفيلد

(Energy Landscape for Hopfield's Autoassociative Network)

وبمقارنة هذه الخطوة بأول خطوة فى الخوارزميات المبينة فى شكل (٣-١٩) يتضح انهما متشابهان، ومن الواضح ان عملية تخزين البصمات تعتمد على تغيير قيم الاوزان فى الجزء المسئول عن كل البصمات ماعدا التى تم تخزينها قبل ذلك، ومن هنا نرى ان ميكانيكية التخزين للبصمة هى فى الحقيقة التقليل لطاقتها.

(٢-٤-١٩) ميكانيكية استدعاء البصمات المخزونة (التذكر)

(Stored Patterns Recall Mechanism)

كما سبق تم عرض ميكانيكية تخزين البصمات فى الشبكة ولنا ان نفرض ان هناك شبكة تحتوى على بصمات مخزونة ويراد استدعاء هذه البصمات مرة ثانية، وتشمل ميكانيكية الاستدعاء او التذكر والتي تعتمد على طريقة انخفاض الميل (Gradient Descent) الخطوات التالية :

١ - اذا رجعنا ثانية الى معادلة دالة الطاقة والمبينة قبل ذلك فى الشكل الآتى:

$$E = -\frac{1}{2} \sum_i \sum_{j \neq i} w_{ij} x_i x_j + \sum_i x_i T_i$$

ونريد ان نقوم بحساب مقدار الطاقة التى يساهم بها العنصر الحسابى الواحد فى دالة الطاقة الكلية ثم العمل على ان يكون هذا الجزء من الطاقة اقل ما يمكن ثم تطبيق ذلك على جميع العناصر حتى نصل الى اقل قيمة لدالة الطاقة الكلية.

٢ - ولتنفيذ ذلك فاننا سوف نقوم بفصل دالة الطاقة الى قسمين، حيث يمثل القسم الاول مقدار مساهمة العنصر (k) فى دالة الطاقة، ويمثل القسم الثانى مساهمة باقى العناصر كما يلى:

$$E = -\frac{1}{2} \sum_{i \neq k} \sum_{j \neq k} w_{ij} x_i x_j + \sum_{i \neq k} x_i T_i$$

$$-\frac{1}{2} x_k \sum_j x_j w_{kj} - \frac{1}{2} x_k \sum_i x_i w_{ki}$$

$$+ x_k T_k$$

٣- اذا فرضنا ان العنصر (k) سوف يغير حالة الخرج له من (x_{k1}) الى (x_{k2}) وبذلك يكون الفرق فى الطاقة $(\Delta E = E_2 - E_1)$ والذي حدث نتيجة الى تغيير الحالة بالمقدار $(\Delta x_k = x_{k2} - x_{k1})$ وبالتعويض فى المعادلة السابقة والطرح يمكن الحصول على الفرق فى الطاقة نتيجة

للتغير فى حالة العنصر (k) فقط كالآتى:

$$\Delta E = -\frac{1}{2} \left[\Delta x_k \sum_j x_j w_{kj} + \frac{1}{2} \Delta x_k \sum_i x_i w_{ki} \right] + \Delta x_k T_k$$

٤- ويمكن استخدام التماثل لتبسيط هذه المعادلة لكى تصبح فى الشكل الآتى:

$$\Delta E = -\Delta x_k \left[\sum_j x_j w_{kj} - T_k \right]$$

والتي تبين ان التغير فى الطاقة الكلية (ΔE) يعتمد على مايلى:

أ - (Δx_k) مقدار التغير فى حالة العنصر (k) .

ب - $\sum_j x_j w_{kj}$ المجموع الموزن للمداخل الى العنصر (k) .

ج- (T_k) القيمة الحدية للعنصر (k) .

ومن المعروف سابقا عند خزن بصمة من البصمات ان يتم وضع القيمة الحدية لكل عنصر مساويا للصفر وذلك للتأكد ان البصمات قد احتلت الآبار التي تمثل اقل قيم للطاقة.

٥ - ولنا ان نتذكر ان قيمة الخرج لكل عنصر اما ان تكون $(+1)$ و (-1) وبذلك فان تقليل قيمة

الفرق فى الطاقة (ΔE_k) لكل عنصر يعني قيم للخرج تساوي $(+1)$ اذا كان المجموع الموزن

اكبر من الصفر وتساوى (-1) اذا كان المجموع الموزن اقل من الصفر وعلى ذلك يمكن كتابة

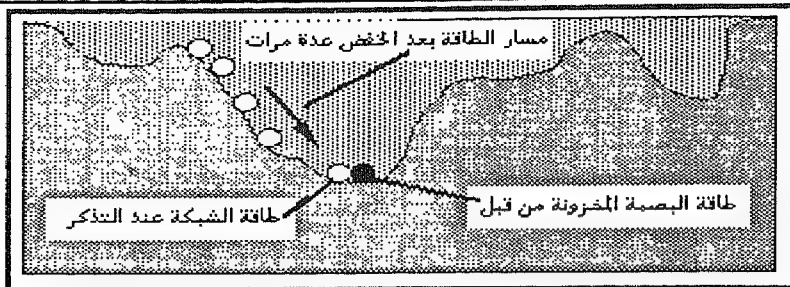
دالة التحديث فى الصورة الآتية :

$$\sum_{j \neq k} w_{ij} x_j \begin{cases} > 0 & x_i & +1 \\ = 0 & \text{remain} & +1 \\ < 0 & x_i & -1 \end{cases}$$

دالة التحديث (Update Function)

وبذلك نرى أن دالة التحديث يمكن لها ان تقوم باجراء خفض للميل والذي يؤدي الى

استرجاع او التذكر لبصمة مخزونة من قبل حيث يتم خفض الطاقة عدة مرات وفى كل مرة تكون



(١٩-٤) خطوات عملية استدعاء البصمة المخزونة (التذكر أو التحديث)
القيمة اقل من السابقة الى ان يتم الوصول الى حالة اتزان ولا يمكن خفض الطاقة بعد ذلك وتعطى الشبكة او تتذكر البصمة المخزونة مرة ثانية، ويمكن شرح ميكانيكية الاستدعاء او التذكر او التحديث كما فى شكل (١٩-٤) حيث يجرى تمثيل البصمة المخزونة من قبل بالكرة السوداء والتي تحتل احد الآبار الذى يمثل اقل طاقة، وتمثل الكرة البيضاء اللون طاقة الشبكة عند بداية عملية الاسترجاع او التذكر. وعند اجراء خفض عدة مرات تنحدر الكرة البيضاء فى مسار الطاقة الاقل الى ان تصل الى التطابق مع الكرة السوداء والتي تمثل البصمة المخزونة وتعرف هذه الخطوة بانتهاء عملية التذكر.

(١٩-٤-٣) التحديث (التذكر) المتزامن والغير متزامن
(Synchronous & Asynchronous Updating)

تنقسم طرق التحديث او التذكر الى:

١ - طريقة التحديث المتزامن (Synchronous Updating) والذي يتم فيها التحديث لكل العناصر مرة واحدة وفى زمن واحد بعد التجميد المؤقت لعمل الدائرة فى حالة معينة وحساب الحالة اللاحقة .

٢ - طريقة التحديث الغير المتزامن (Asynchronous Updating) والذي يتم فيها التحديث لعنصر واحد يتم اختياره عشوائيا، حيث يعتمد الخرج الجديد على الدخل الذى يستقبله العنصر ثم تكرر هذه العملية باختيار عنصر اخر وهكذا. ويمكن الاختلاف الاساسى بين الطريقتين ان قيمة الخرج للعنصر المحدث فى الطريقة الثانية سوف تؤثر على باقى العناصر، وبذلك يكون لها تأثير على حالة الشبكة والتي تعنى ان ترتيب العناصر المختارة عشوائيا سوف تؤثر على السلوك العام للشبكة ككل والذي لا يوجد فى الحالة الاولى حيث يتم التحديث لكل فى نفس الوقت.

وبوضح المثال المبين شكل (١٩-٥) كيف تم تعليم الشبكة على مجموعة الارقام العربية من نمط معين وبعد التعليم تم تعريض الرقم (٤) من نمط آخر والذي يختلف عن النمط الاول عند مدخل الشبكة وبعد عدة دورات تتم عملية التذكر او الاستدعاء وتعطى الشبكة الرقم

٥	٤	٣	٢	١
٠	٩	٨	٧	٦
مجموعة الأرقام العربية التي تعلمت عليها الشبكة وتم تخزينها بها				
ع	الرقم ٤ عند المخرج	•	الرقم ٤ عند المدخل	ع

(١٩-٥) عملية استدعاء البصمة المخزونة باستخدام بصمة مشوشة

المخزون الاصلى. ولقد وجد عمليا ان اكبر عدد من البصمات التى يمكن تخزينها فى الشبكة بدون تداخل بينها هو (0.15 N) حيث (N) هو عدد العناصر الحسابية بالشبكة والتى تعنى ان شبكة مكونة من ١٠٠ عنصر تكون مناسبة لحزن ١٥ بصمة فقط بدون تداخل بينها.

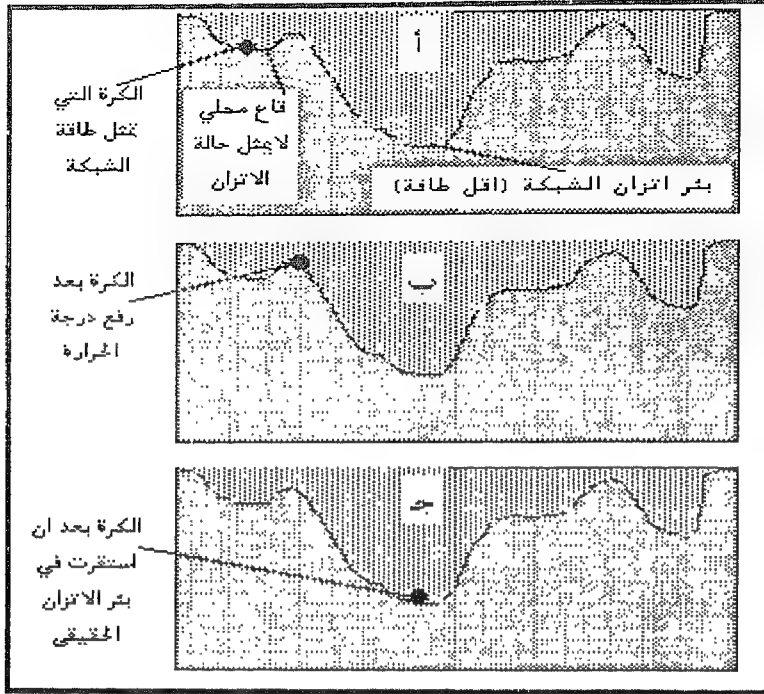
(١٩-٥) آلة بولتزمان الحرارية (The Boltzmann Thermal Machine)

كما سبق يتضح ان ايجاد الحل لشبكات هوفيلد هو الوصول الى اقل مستوى للطاقة والذي يقابل احتلال الكرة (التي تمثل طاقة الشبكة) القاع ذو الطاقة الاقل على الاطلاق (بشر الحل الامثل) فى مسطح مستويات الطاقة، ونظرا لوجود قيعان محلية تكون فيها الطاقة قليلة ولكنها ليست الحل الامثل (Optimal) كما هو موضح فى شكل (١٩-٦-أ) حيث تصل الشبكة الى قاع لايمثل الحل ولكى نتأكد من ذلك فاننا نقوم برفع درجة حرارة الكرة وذلك بزيادة الطاقة الحرارية لها فجأة او بإضافة ضوضاء، وبذلك تتحرك الكرة عشوائيا الى اعلى كما فى شكل (١٩-٦-ب) ثم تترك لتبرد ببطء وذلك بواسطة تقليل درجة الحرارة وينشأ من ذلك احد وضعين:

الوضع الاول : وهو رجوع الكرة الى نفس القاع اذا لم يكن قاعا محليا وبذلك يصبح هذا القاع هو الممثل للحل الامثل واقل طاقة ممكنة.

الوضع الثانى: وهو انزلاق الكرة الى البئر الحقيقية التى تمثل ادنى مستوى للطاقة والحل الأمثل كما فى شكل (١٩-٦-ج).

ويمكن تقثيل آلة بولتزمان على انها محاكاة لعملية التخثير الحرارى للمعادن (Thermal Annealing) حيث يسخن المعدن الى ان ينصهر ثم يترك ليبرد ببطء حتى يصل الى اقل طاقة ممكنة. وتعتبر آلة بولتزمان الحرارية هى الجمع بين خوارزم هوفيلد للتعلم مع استخدام قاعدة التحديث المعتمدة على نظرية الاحتمالات (Probabilistic Update) حيث يقوم كل عنصر حسابى فى الشبكة بحساب الحالة التى يمكن ان ينتقل اليها فى اتجاه تقليل الطاقة وذلك



شكل (١٩-٦)

باستخدام قاعدة الاحتمالات والذي يعنى انه في بعض الاحيان تقفز الشبكة الى احتلال طاقات اعلى بحيث يمكنها ان تقفز من القيعان المحلية ويمكن اختيار دالة الاحتمالات بحيث تسمح للشبكة بالتغيير الى حالة الطاقة الاقل اذا كانت هناك فائدة من ذلك، وعند الوصول الى الاحتمالية تساوى (١) عند تخفيض درجة الحرارة فانه يمكن القول بان الشبكة قد وصلت الى حالة الاتزان الحرارى (Thermal Equilibrium) ويمكن استخدام الرياضيات لتوضيح ذلك كمايلي :

١ - يقوم كل عنصر حسابى فى الشبكة بحساب مقدار فجوة (ΔE_k) للطاقة فى الشكل :

$$\Delta E_k = \sum_i w_{ki} s_i - \Theta_k$$

٢- ينتقل كل عنصر من حالته الى الحالة التى تمثل الطاقة الاقل وذلك باستخدام قاعدة التحديث الاحتمالية (Probabilistic Update) وذلك بمقدار احتمالية (P_k) فى الصورة :

$$P_k = \frac{1}{1 + e^{\frac{-\Delta E_k}{T}}}$$

٣ - وبذلك يمكن للشبكة ان تحتل العديد من مستويات الطاقة التى يحكمها توزيع بولتزمان (Boltzmann Distribution)، فاذا فرضنا ان (P_α) هى الاحتمالية التى تكون عليها حالة عناصر الشبكة فى مستوى الطاقة (E_α) طبقا للتوزيع البولتزمانى فى الشكل الآتى:

$$P_{\alpha} = k e^{\frac{-E_{\alpha}}{T}}$$

وان (P_{β}) هي الاحتمالية التي تكون عليها حالة عناصر الشبكة فى مستوى الطاقة (E_{β}) وذلك طبقا للتوزيع البولتزمانى فانه يمكن كتابة مايلى:

$$\frac{P_a}{P_b} = \frac{e^{\frac{-E_a}{T}}}{e^{\frac{-E_b}{T}}} = e^{\frac{-(E_a - E_b)}{T}}$$

واذا فرضنا ان (E_{α}) هي الطاقة الادنى وان (E_{β}) هي الطاقة الاعلى وعند التوازن الحرارى للشبكة عندما تثبت قيم الاحتمالات لكل مستوى للطاقة فاننا يمكن ان نقرر مايلى:

$$\begin{array}{ccc} E_{\alpha} & < & E_{\beta} \\ e^{-(E_{\alpha} - E_{\beta})} & > & 1 \end{array}$$

$$\text{Therefore } \frac{P_{\alpha}}{P_{\beta}} > 1$$

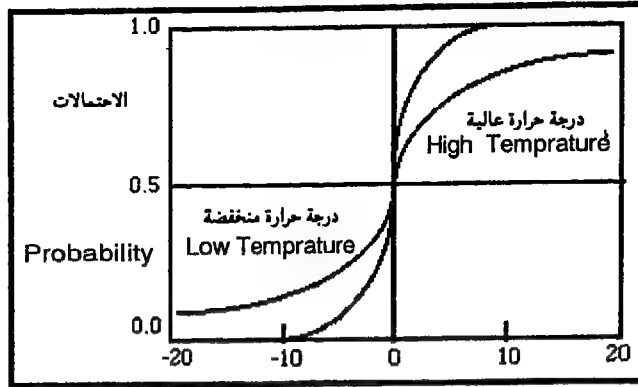
$$\text{So } P_{\alpha} > P_{\beta}$$

والتي تعنى انه عند التوازن الحرارى تكون نسبة الاحتمالات لوجود مستويات من الطاقة المنخفضة اكثر من نسبة الاحتمالات لوجود مستويات من الطاقة المرتفعة. وتصل الشبكة الى التوازن عند درجات الحرارة العالية بسرعة اكثر من الدرجات المنخفضة ولكن يكون ذلك على حساب مستويات الطاقة.

ان عملية تخفيض الطاقة بينما تعمل الشبكة والتي تعتبر محاكاة لعملية التخمير الحرارى تسمح بالوصول بسرعة الى توازن حرارى عند الطاقة المنخفضة، بينما تدفع الحرارة العالية الشبكة لتجنب القيعان المحلية، وعند الاتزان الحرارى للطاقات المنخفضة تكون قيم الاحتمالات ثابتة، ويبين شكل (٧-١٩) تأثير درجة الحرارة على دالة احتمالات الانتقال.

ويجرى تغيير درجات الحرارة بضبط درجة الانحناء للدالة المقطعية والتي تحدد بشكل قاطع احتمال ان يحتل عنصر ما مستوى الطاقة الطبيعى المناسب له والحالى من الضوضاء، فاذا كانت القيمة اكبر من القيمة الحدية بكثير فان قيمة الاحتمال يكون (١) واذا كانت القيمة اقل من القيمة الحدية فان الاحتمال يبقى صفرا دائما.

ويعتبر معدل خفض درجة الحرارة من العوامل الاساسية حيث أنه يساعد على وصول



$$P_K = \frac{1}{1 + e^{\frac{-\Delta E_i}{T}}} \quad (٧-١٩) \quad \text{تأثير درجة الحرارة على احتمالات دالة الانتقال}$$

والتي توضح ان الاحتمال للانتقال الى طاقات عالية اكبر عند درجات الحرارة العالية بالمقارنة عند الدرجات المنخفضة.

الشبكة للحل الأمثل، فإذا كان معدل الخفض سريعاً جداً فإن ذلك يؤدي إلى وقوع الشبكة في قيعان محلية بعيدة عن الحل الأمثل، وعلى العكس إذا كان المعدل بطيئاً فإن الشبكة تكون قادرة على الهروب من القيعان المحلية ولكنها تصل إلى الحل الأمثل بعد مدة كبيرة.

(١-٥-١٩) خوارزم التعلم لآلة بولتزمان الحرارية

(The Boltzmann Thermal Machine Learning Algorithm)

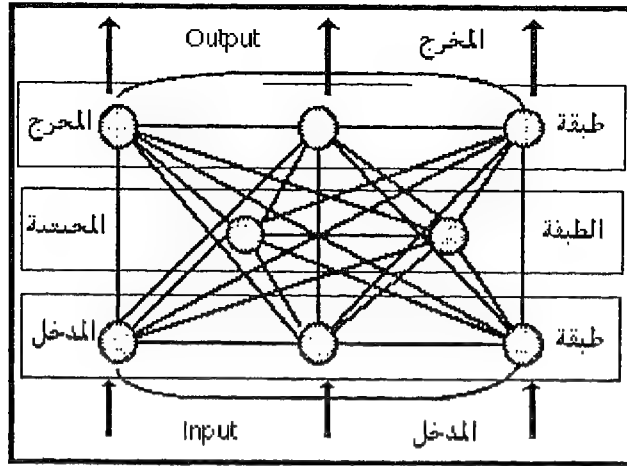
يمكن وضع شبكات هوبفيلد الكاملة الاتصال في شكل الشبكات متعددة الطبقات بحيث يكون هناك طبقة للمدخل وطبقة للمخرج وطبقة مختفية كما في شكل (٨-١٩)، وينقسم خوارزم التعلم لآلة بولتزمان الحرارية إلى طورين :

١ - الطور الأول (First Phase) :

ويجرى تنفيذ هذا الطور بتقييد وتثبيت المداخل والمخارج عند القيم المعينة والصحيحة، ثم تترك الشبكة لتعمل عدة دورات بداية من درجة الحرارة الابتدائية مع تخفيض درجة الحرارة تدريجياً إلى أن تصل الطبقة المختفية إلى حالة التوازن الحراري (Thermal Equilibrium) ثم يجرى تطبيق نظرية التعلم الهيبباني (Hebbian Learning) على كل عنصرين حسابيين متصلين ببعض ولهم نفس الحالة النشطة (on) وذلك بزيادة الوزن بقيم صغيرة في كل مرة (Weights are incremented).

١ - الطور الثاني (Second Phase) :

وفي هذا الطور يتم تثبيت المداخل فقط عند القيم المعينة والصحيحة وتترك طبقة المخرج والطبقة المختفية في شكل حر بدون التقيد بقيم ثم تترك الشبكة لتعمل عدد من الدورات مع



(١٩-٨) تمثيل لشبكة هوبفيلد ذات الاتصال الكامل في شكل شبكة متعددة الطبقات

تخفيض درجة الحرارة تدريجياً إلى أن تصل إلى التوازن الحراري ثم يجري تخفيض قيم الوزن بين كل عنصرين حسابيين متصلين ببعض ولهم نفس الحالة النشطة (on) بقيم صغيرة في كل مرة (Weights are decremented). ثم يجري تكرار الطور الأول متبوعاً بالطور الثاني عدة مرات إلى أن تثبت قيم الأوزان. ويمكن القول بأنه يجري تقوية العناصر النشطة في الطور الأول بينما يجري تقليل مساهمة هذه العناصر في الطور الثاني والذي يؤيد النظرية الهيبيانية للتعليم. ويمكن تلخيص أهم المزايا لآلة بولتزمان مايلي:

- ١ - يمكن للشبكة الوصول إلى آبار التوازن بسرعة حيث أنه يمكنها الهروب من القيعان المحلية نتيجة لاحتلالها طاقات أكبر بمقدار احتمالية يعتمد على طاقة العنصر.
- ٢ - تحتل الشبكة أفضل آبار التوازن والذي يعتبر في الحقيقة أفضل عملية تخزين وبذلك تعتبر الشبكة في وضع مناسب للاستدعاء والتذكر.
- ٣ - يمكن استخدام الشبكة لحزن الاحتمالات حيث تتناسب قيمة الاحتمالية مع الطاقة لكل عنصر. ويشتمل الخوارزم لآلة بولتزمان على الخطوات التالية :

أ - الطور الأول (طور التقوية)

- ١- يتم اختيار عناصر لطبقتي المدخل والمخرج.
- ٢- يتم ربط وتثبيت قيم المداخل والمخارج بالقيم الصحيحة المعينة.
- ٣- يتم حساب الطاقة لجميع الحالات في الشكل :

$$\Delta E_k = \sum_i w_{ki} s_i - \Theta_k \quad 0 \leq i \leq N-1$$

٤ - يتم الانتقال الي حالات اقل من الطاقة باستخدام الاحتمالية في الصورة:

$$P_K = \frac{1}{1 + e^{\frac{-\Delta E_k}{T}}}$$

٥ - يجرى تخفيض درجة الحرارة (T) عدة مرات الى ان تثبت قيمة المخرج.

٦ - يجرى زيادة الوزن بين كل عنصرين اذا كانت حالتها نشطة.

ب - الطور الثاني (طور الاضعاف)

١- يتم ربط وتثبيت قيم المداخل فقط بالقيم الصحيحة المعينة مع ترك الطبقة المختفية وطبقة المخرج حرتين.

٢- تترك الشبكة للقيام بعمل الحسابات كما فى الطور الاول حتى تصل الى حالة الاتزان الحرارى.

٣- يجرى تقليل او اضعاف الوزن بين كل عنصرين اذا كانت حالتها نشطة.

ج - يجرى تكرار العملية عدة مرات على التوالى الى ان تصبح الازنان فى حالة من الاتزان.

(١٩-٦) بعض التطبيقات لآلة بولتزمان الحرارية

(Some Application for Boltzmann Thermal Machine)

(١٩-٦-١) مشكلة البائع المتجول (The Travelling Salesman Problem)

يمكن استخدام آلة بولتزمان الحرارية لوضع الحول للمشكلات الصعبة والمركبة التى تحتوى على عدد من المتطلبات والقيود المتعارضة، حيث يتطلب حلها تنفيذ كثير من الشروط. ومن هنا فان اختيار مشكلة البائع المتجول يعتبر إختبار لقدرة الشبكة العصبية لبولتزمان على ايجاد الحل لها، حيث انها تشتمل على كثير من المتطلبات والقيود والشروط المتعارضة.

تتلخص المشكلة فى ايجاد أقصر المسارات التى من المفروض ان يقطعها البائع المتجول لكى يزور عدد من المدن المختلفة التى تقع على مسافات مختلفة، على ان تنتهى الزيارة من المدينة التى بدأ البائع منها جولته، على ان لا تتكرر زيارة أى مدينة أكثر من مرة واحدة.

وتعتبر الشروط او القيود التى يجب تنفيذها هى:

١- ان يتم زيارة المدينة مرة واحدة فقط.

٢- ان تكون المسافة خلال المسار بين أى مدينتين اقصر مما يمكن .

من الواضح ان اختيار دالة للطاقة تاخذ فى الاعتبار هذين الشرطين سوف تعطى الحل الأمثل للشبكة عند تخفيض هذه الطاقة الى اقل قدر ممكن.

ولايجاد الحل للمشكلة لنا ان نتبع الخطوات التالية:

١ - وضع المشكلة بشكل يسهل تمثيلها فى الشبكة وذلك بايضاح اسم المدينة وترتيبها فى الزيارة. فاذا فرضنا عدد (n) من المدن فاننا سوف نفرض ان المدينة الواحدة يمكن تمثيلها بعنصر حسابى واحد ولذلك يتطلب وجود عدد (n) من العناصر الحسابية .

٢ - فى حالة زيارة مدينة واحدة فى الزمن الواحد فان العنصر الحسابى المناظر لها لابد ان يكون نشطا (on) وان تكون العناصر الاخرى الممثلة للمدن الاخرى خامدة (off)، فاذا فرضنا انه يوجد اربعة مدن فقط هي:

(A,B,C,D)

وانه سوف يتم زيارة المدينة (A) اولا فان ذلك يحتم ان يكون العنصر الاول نشطا (ON) ويعطى واحد فى الخرج وجميع العناصر الاخرى خامدة (off) وتعطى خرجا يساوى الصفر، ويمكن تمثيل ذلك فى الصورة 1000، وعلى ذلك فاننا نحتاج الى مصفوفة مكونة من ١٦ عنصر (اربعة فى اربعة) لتمثيل اسم المدينة ومكان كل منها كما هو مبين فى المثال شكل (١٩-٩)، حيث يبين انه يمكن زيارة المدن بالترتيب الآتى:

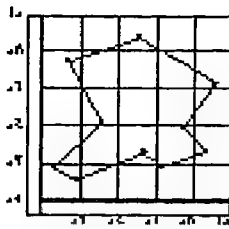
(A) ثم (D) ثم (B) ثم (C).

٣ - يجب ان تحتوى دالة الطاقة على القيود الآتية:

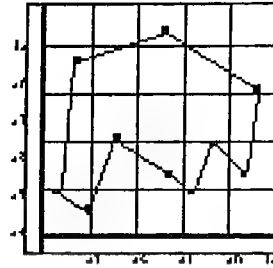
أ - يحتوى كل صف من المصفوفة على (1) فقط والذي يبين ان المدينة تم زيارتها مرة واحدة.
ب - ان يحتوى كل عمود من المصفوفة على (1) فقط والذي يبين ان المدينة تم زيارتها.
ج - ان تحتوى المصفوفة على عدد (n) من الواحدات (1) والذي يبين ان الزيارة تكون

المكان	1	2	3	4
المدينة				
A	1	0	0	0
B	0	0	1	0
C	0	0	0	1
D	0	1	0	0

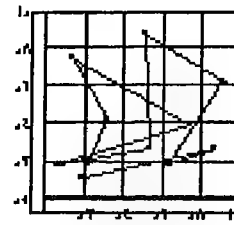
(١٩-٩) مثال للمصفوفة التى تمثل اسماء المدن ومكانها



شكل (١٩-١) ج مسار يمثل
شكل آخر من الحل الأمثل
باستخدام الشبكة



شكل (١٩-١) ب
مسار يمثل الحل الأمثل بين
المدين باستخدام الشبكة



شكل (١٩-١) ج
مسار عشوائي بين
مدن بدون استخدام الشبكة

(١٩-١٠) استخدام شبكة (آلة) بولتزمان لايجاد اقصر المسارات

التي يمكن للبائع المتجول ان يسلكها

صحيحة وواقعية.

٤ - نفرض انه يمكن كتابة قيمة الخرج للشبكة في الشكل الآتي:

الخرج V_X حيث X يمثل اسم المدينة و I تمثل مكانها.

وبذلك يمكن كتابة دالة الطاقة في الشكل الآتي:

$$E = A \sum_X \sum_I \sum_{j=1}^n V_{Xj} V_{Xj} + B \sum_I \sum_X \sum_{X=Y} V_{Xj} V_{Yj} + C \left(\sum_X \sum_I V_{Xj} - n \right)^2$$

$$+ \frac{1}{2} D \sum_X \sum_I \sum_{Y=X} d_{XY} V_{Xj} (V_{Yj+1} + V_{Yj-1})$$

حيث يمثل الجزء الاول القيد المفروض على اسم المدينة X ويصبح هذا الجزء مساويا للصفر اذا كان كل صف والمبين لاسم للمدينة X يحتوى على الرقم واحد فقط. ويمثل الجزء الثانى القيد المفروض على مكان المدينة في المسار ويساوى الصفر اذا كان العمود الممثل لمكان المدينة في المسار يحتوى على الرقم واحد فقط. ويمثل الجزء الثالث القيد المفروض على ان احتواء المصفوفة على عدد (n) من الواحدات (1) والذي يبين ان الزيارة تكون صحيحة وواقعية. ويمثل الجزء الرابع القيد المفروض على المسافة لتكون أقصر مايمكن حيث تمثل

المسافة (d_{XY}) بين (X) و (Y) ويمثل الجزء الذى يلى (d_{XY}) قيمة لاتساوى الصفر اذا كانت المدينة (X) والمدينة (Y) متتاليتين وتصبح القيمة صفرا لغير ذلك ويصبح المجموع ممثلا لطول المسار الكلى لجولة البائع.

ويبين شكل (١٩-١٠) كيف يمكن الحصول على الحل الأمثل لأقصر المسارات بين المدن.

٥ - تحتوى دالة الطاقة على الثوابت (A, B, C, D) والتي يتم اختيارها عشوائيا وذات قيم كبيرة ويصبح تخفيض الطاقة الى ادنى مستوى هو فى الحقيقة الوصول الى الحل الامثل ليعطى اقصر المسارات على الاطلاق اذا احتلت دالة الطاقة للبئر الممثل للحل الأمثل.

الفصل العشرون

نظرية الرنين المتكيف
للشبكات العصبية
المكثفة التوازي

**Adaptive Resonance Theory
(ART) Massively Parallel
Neural Networks**

(٢٠-١) نظرية الرنين المتكيف (Adaptive Resonance Theory (ART))

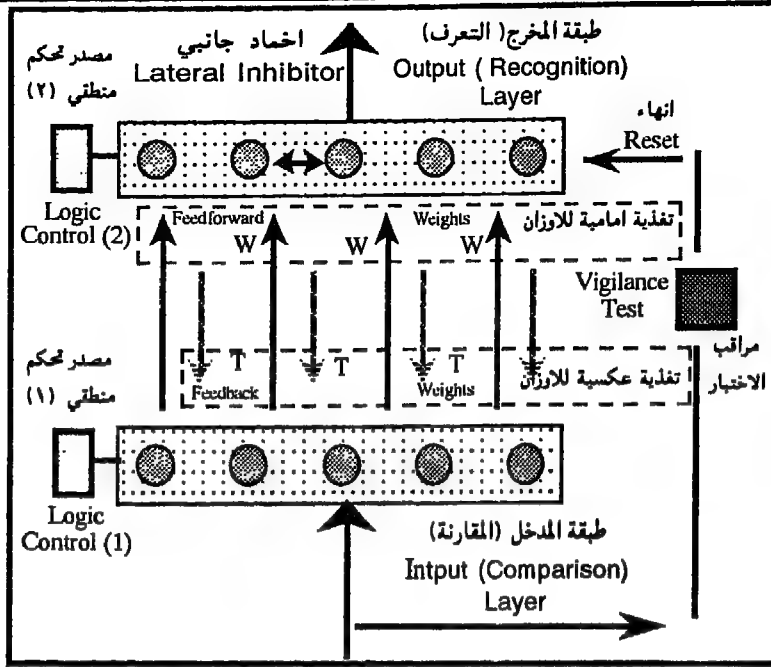
قام جروسبرج وجيل كاربنتر (Grossberg & Gail Carpenter) بتطوير هذه الشبكات اعتماداً على نمذجة ميكانيكية العمل للمخ الانساني في شكل نظم ذاتية التنظيم (Self-organizing Systems) ومحاولة للتغلب على جمود الشبكات السابقة التي لا يمكن لها قبول تعلم بصمات جديدة فوق البصمات المخزونة، وتشير كلمة الرنين الى التشابه الذي يحدث بين تردد المتجهات للاوزان بين طبقتي المدخل والمخرج عدة مرات قبل ان تستقر وما يحدث في الدوائر الالكترونية المهتزة من رنين للموجات الكهربية. تعتبر هذه الشبكات والمعتمدة على نظرية الرنين المتكيف احدى دعائم تقنيات الحساب العصبى وذلك لما يلي:

- ١- تحتوى معمارية الشبكة على تكثيف للتوصيلات المتوازية بين العناصر والتي تساعد على اجراء المعالجة المتوازية .
- ٢ - تعتبر نمذجة لميكانيكية العمل التي يتبعها المخ الانساني في سرعة التعرف على الاشياء والبصمات بالرغم من كبر حجم قاعدة المعلومات التي يتم البحث خلالها .
- ٣ - يمكن للشبكة التعرف السريع على البصمات المخزونة في طبقة التعرف والتي تستخدم فيها الطرق المثلى للبحث المتوازي في طبقة التعرف.
- ٤ - يجرى التوصيف للمتجهات المدخلة باستخدام تقنية متوازية حيث يتم مقارنته بالمتجهات المخزونة في عناصر طبقة التعرف في نفس الوقت، واذا لم يكن مشابهاً لأى منها فإنه يجرى تخزينه في عنصر جديد ويجرى تتابع هذه العملية الى ان تقتل الذاكرة. تعتبر هذه الخاصية من اهم خواص هذه الشبكات والتي تتميز بها عن الشبكات السابق ذكرها.

(٢٠-٢) معمارية شبكات نظرية الرنين المتكيف

(ART Network Architecture)

- يبين شكل (١-٢٠) معمارية الشبكات نظرية الرنين المتكيف والتي تتكون من الآتى:
- ١ - طبقة اولى للدخال (Input Layer) يطلق عليها طبقة المقارنة (Comparison Layer) وطبقة ثانية للإخراج (Output Layer) يطلق عليها طبقة التعرف (Recognition Layer) .
 - ٢ - يجرى استخدام الاتصال المتوازي المكثف بين العناصر في الطبقتين، حيث يتصل كل عنصر في طبقة بجميع العناصر الأخرى في الطبقة الثانية بوصلتين كما في شكل (٢-٢٠) حيث تمثل الوصلة الاولى الاتصال للتغذية الأمامية للاوزان (Feedforward Weights)



شكل (٢٠-١) معمارية شبكات الرنين المتكيف

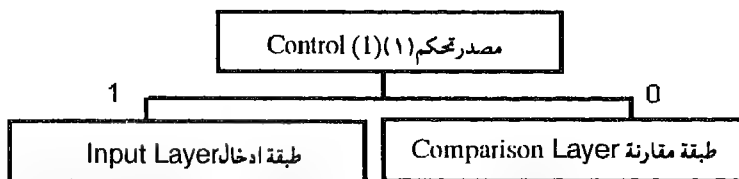
والذي يمثلها المتجه (W) والوصلة الثانية ومثل التغذية العكسية (Feedback Weights) من المخرج الى المدخل ويمثلها المتجه (T) كما يوجد اتصال جانبي للاخاد (Lateral Inhibitor) بين كل عنصر والعنصر الذي بجانبه في طبقة الاخراج.

٣ - يجرى تزويد كل طبقة بمجموعة من الدوائر المنطقية والتي تعطي اشارات منطقية للتحكم (Logic control signals) في تدفق البيانات في كل طبقة على حدة وذلك اثناء دورة العمل، ويمكن تسمية هذه الاشارات بالتحكم (١) (Control(1)) والخاص بطبقة المدخل والتحكم (٢) (Control(2)) والخاص بطبقة المخرج.

٤ - تحتوي الشبكة على مراقب الاختبار (Vigilance Test) والذي يلعب دورا في منتهى الاهمية بين طبقة المدخل وطبقة المخرج، حيث يتولى مقارنة القيم المدخلة الى الطبقة الاولى بقيمة حدية اختبارية (Vigilance Values) حيث يمكن تقرير العمل دورة اخرى لايجاد قيم جديدة للخروج، وعلى ذلك يقوم بمسح او انهاء (Reset) القيم السابقة الموجودة في المخرج.

وما سبق يتضح السمات الآتية لهذه الشبكات:

- أ - التغذية العكسية المكثفة .
- ب- انفراد كل طبقة بعمل معين .
- ج - التحكم الخارجى باشارات منطقية.



شكل (٢٠-٢) الطور الاول من إشارات التحكم المنطقية

(٢٠-٣) ميكانيكية العمل لشبكات نظرية الرنين المتكيف

(Operation Mechanism For ART Networks)

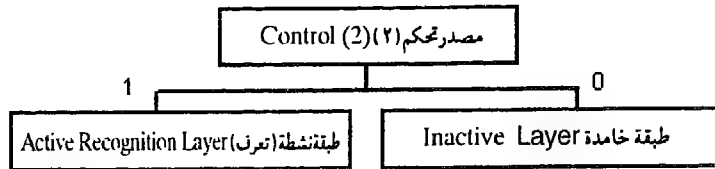
تختلف هذه الشبكات عن الشبكات السابقة في التركيب وميكانيكية العمل حيث يعتبر الفرق الاساسى هو ظاهرة الرنين (Resonance) والتي تحدث بشكل مماثل لما يحدث فى دوائر الرنين الكهربية والالكترونية حيث يقوم متجه الوزن (W) بالانتقال من طبقة المدخل الى طبقة المخرج ثم ينعكس فى شكل متجه آخر (T) عن طريق التغذية العكسية من طبقة المخرج الى طبقة المدخل ثم يكرر هذه العملية عدة مرات بعد تعديلته فى كل مرة. وتشتمل ميكانيكية العمل لهذه الشبكات على الأطوار الآتية :

- ١- طور التنشيط (Initialisation Phase).
- ٢- طور التعرف (Recognition Phase).
- ٣- طور المقارنة (Comparison Phase).
- ٤- طور البحث (Search Phase).

(٢٠-٣-١) طور التنشيط (Initialisation Phase)

يختلف طور التنشيط فى هذه الشبكات عن التنشيط المعروف سابقا للشبكات الاخرى حيث تقوم مصادر اشارات التحكم المنطقية (١) و (٢) شكل (٢٠-٢) و (٢٠-٣) بالعمل لتوجيه البيانات داخل الشبكة عند اجراء عملية التعليم وكذلك عند التوصيف. وتعمل اشارات التحكم المنطقية (١) فى تنظيم عمل طبقة الادخال كما يلى:

- أ - تاخذ قيمة اشارة التحكم الرقم الثانى (١) اذا كان هناك قيم صحيحة للمدخل (لاتساوى الصفر) وتعمل الطبقة كطبقة ادخال.
- ب - تاخذ قيمة اشارة التحكم الرقم الثانى (صفر) اذا كانت احدى عناصر طبقة الاخراج نشطة. وتعمل اشارات التحكم (٢) فى تنظيم عمل طبقة الاخراج كما يلى:
- ١ - تاخذ قيمة اشارة التحكم الرقم الثانى (١) اذا كان هناك قيم صحيحة للمدخل (لا تساوى الصفر) وتعمل الطبقة كطبقة تعرف.



شكل (٢-٣) الطور الثاني من إشارات التحكم المنطقية

٢ - تأخذ قيمة إشارة التحكم الرقم الثنائي (صفر) إذا فشلت عملية مراقبة الاختبار وفي هذه الحالة يتم انهاء العمل بهذه الطبقة ويتحول نشاط العناصر الى الصفر.

يجرى تنشيط الموجهات للاوزان الامامية والعكسية كمايلي :

أ - ياخذ المتجه العكسي للوزن (T) الرقم الثنائي (١) وذلك مشيرا الى اتصال كل عنصر من طبقة الاخراج بكل عنصر فى طبقة الادخال.

ب - ياخذ المتجه الامامى للوزن (W) قيمة حقيقية ثابتة تساوى :

$$w_i = \frac{1}{1+n}$$

حيث (n) تمثل عدد العناصر الحسابية فى طبقة الادخال ، مع وضع حد المراقبة

فى المدى الآتى: $0 < p < 1$

(Recognition Phase) طور التعرف (٢-٣-٢٠)

يبدأ هذا الطور بالسماح بارسال المتجه المدخل من طبقة الادخال الى طبقة الاخراج حيث يتم مقارنة قيمته بالقيمة للمتجه الذى يوصف البصمة السابقة والمخزونة فى كل عنصر من عناصر الاخراج وذلك بهدف الوصول الى احسن قيمة للمواءمة بينهما. وفيمايلي شرح كيف يمكن لعناصر هذه الطبقة من تنفيذ ذلك اثناء دورة التعلم، وبين شكل (٢-٤) ان كل عنصر من عناصر طبقة الادخال يتصل به ثلاث اطراف للمدخل هي:

١ - مركبة المتجه المُدخل (x_i) .

٢ - متجه التغذية العكسية من طبقة المخرج الي طبقة المدخل (T_1).

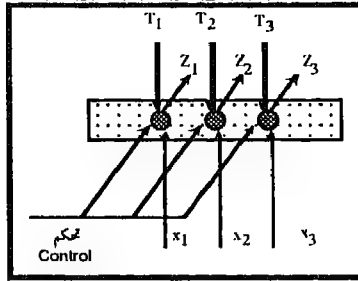
٣ - إشارة التحكم من مصدر التحكم المنطقى (١).

ويتصل العنصر بطرف واحد للخروج يحمل متجه المقارنة (Comparison Vector) ويمكن

القول بان تدفق البيانات خلال طبقة المدخل تسير طبقا لقاعدة الثلثين (Two-third Rule)

والتي تم اقتراحها من قبل جروسبرج وكارينتر والتي تنص على مايلي:

" يكون خرج العنصر نشطا اي يساوي واحد اذا كان اثنين من الثلاث قيم للمداخل نشطة ويكون صفرا في غير ذلك. وهناك تشابه بين طريقة التعلم



شكل (٢٠-٤) قاعدة الثلاثين لجروسبيرج

لهذه الشبكات وشبكات كوهان ذاتية التنظيم "

وتتلخص عملية التعرف فى الخطوات الآتية:

١ - يمكن تصور ان كل متجه للوزن (W) عند كل عنصر للتعرف هو عبارة عن كمية مخزونة تمثل جزء من البصمة المعرفة قبل ذلك.

٢ - يتم مقارنة المتجه المدخل مع المتجه المخزون فى كل عنصر من عناصر طبقة الاخراج وذلك للوصول الى افضل وضع للمواءمة (Best-match) بينهم .

٣ - يجرى الوصول الى افضل وضع للمواءمة (Best-match) وذلك بحساب حاصل الضرب النقطى (Dot Product) بين المتجهين لكل عنصر (متجه المدخل والمتجه المخزون بالعنصر) وتعتبر اكبر قيمة هى الدليل على ان العنصر هو افضل العناصر الموائمة لمتجه الوزن المدخل.

٤ - ومن الطبيعى ان يكون هناك اكثر من عنصر من العناصر يكون حاصل الضرب النقطى قريب من القيمة العظمى ، وللوصول الى عنصر واحد فقط فان ذلك يتم بالاستعانة بالاخماد الجانبي (Lateral Inhibition) حيث يقوم كل عنصر بمحاولة اخماد العنصر الذى يقع بجانبه اذا كانت قيمة حاصل الضرب النقطى للاخير اقل من القيمة العظمى وفى المقابل فان كل عنصر يحاول ان يقوى مركزه (Reinforcement) وذلك بالاستعانة بالتغذية العكسية الموجبه (Positive Feedback) لكل عنصر مع نفسه والتى تعمل لتقوية ولزيادة قيمة حاصل الضرب وتكون المحصلة لهاتين القوتين المتضادتين هو الوصول الى عنصر نشط واحد فقط يطلق عليه العنصر الفائز (Winning Node).

٥ - من المطلوب فى هذه المرحلة ان يقوم العنصر الفائز بنقل قيمة المتجه (T)(Class Ex- (amplifier) المخزون به والذي يعرف بالمتجه المثل لرتبة البصمة الى الطبقة الاولى عن طريق التغذية العكسية وذلك للقيام بعقد المقارنة، فاذا فرضنا ان هذا المتجه قد تم تخزينه فى الشكل الثنائى كما هو موضح فى شكل (٢٠-٥) فى العنصر الفائز فان نقله الى الطبقة المقارنة يتم بسهولة وذلك عن طريق نقل القيم الى العناصر بهذه الطبقة بالتغذية العكسية.

(٢٠-٣-٣) طور المقارنة (Comparison Phase)

كما هو مبين فى شكل (٢٠-٤) من وجود ثلاث قيم عند مدخل كل عنصر من عناصر الطبقة الاولى والتي سوف يتم فيها المقارنة وهذه القيم هى :

- مركبة المتجه المدخل (x_i) .

- ومتجه التغذية العكسية من طبقة المخرج الى طبقة المدخل (T_1) .

- واسارة التحكم من مصدر التحكم المنطقى (١) .

وحيث انه يتم اختيار اسارة التحكم تساوى الصفر لكى تعمل الطبقة كطبقة للمقارنة، ويتطبيق قاعدة الثلاثين فان العنصر سوف يعطى خرجا يساوى المتجه (z) والذي يعرف بمتجه المقارنة والذي يتم ارساله مع المتجه المدخل الى دائرة الانهاء التى تتحكم فى طبقة التعرف.

(٢٠-٣-٤) حد مراقبة الاختبار (Vigilance Threshold Test)

تقوم دائرة مراقب الاختبار باختبار التشابه بين متجهين : متجه المدخل (x_i) ومتجه المقارنة

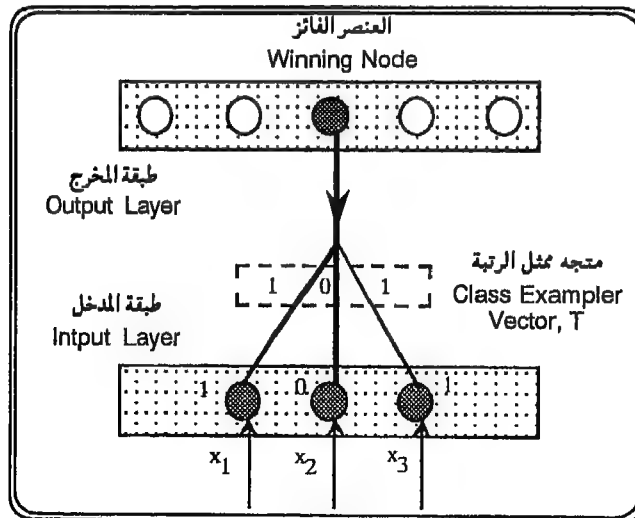
(z) ، وذلك بمقارنة كل منهم على حدة بالقيمة الحدية للمراقبة $(\rho \text{ Vigilance Threshold})$

ويعتبر الاختبار هو تحديد نسبة عدد الواحدات (١) فى كل من متجه المدخل (x_i) ومتجه المقارنة

(z) (حاصل ضرب نقطى للمتجهين) بالنسبة الى عدد الواحدات فى متجه المدخل (x_i) وذلك

لتعطى النسبة (S) كما يلى:

$$S = \frac{\sum t_{ij} x_i}{\sum x_i}$$



شكل (٢٠-٥) نقل متجه العنصر الفائز الى العناصر الاخرى بالتغذية العكسية

ويكون الاختبار فى الصورة الآتية:

إذا كانت (S) اكبر من القيمة الحدية للمراقبة (p) فان التوصيف يكون كاملا ويصبح العنصر الفائز فى طبقة التعرف هو العنصر النشط الواحد فى هذه الطبقة الذى يمثل البصمة، وإذا كان العكس فانه يجرى العمل مرة ثانية للوصول الى افضل قيمة للموائمة وتدخل الشبكة طور البحث.

(٢٠-٣-٥) طور البحث (Search Phase)

فى هذا الطور تقوم الشبكة بمحاولة ايجاد افضل مواءمة بين المنتجه المخزون فى طبقة التعرف والمنتجه المدخل فى طبقة المقارنة وذلك بناءً على نتيجة مراقب الاختبار والخطوات التالية :

١ - يتم تحويل العنصر النشط فى طبقة التعرف (العنصر الفائز) الى الخمود بحيث تعطى خرجاً يساوى الصفر، وبذلك لا يدخل هذا العنصر مرة اخرى فى المقارنة او التنافس للوصول الى افضل مواءمة مرة اخرى مع تحول اشارة مصدر التحكم الى الصفر.

٢ - يتم ادخال المنتجه الجديد الى طبقة التعرف، ويجرى الوصول الى افضل مواءمة وذلك باتباع الخطوات المذكورة فى طور التعرف والوصول الى العنصر الفائز مرة اخرى.

٣ - يتم ادخال الشبكة فى طور المقارنة مرة اخرى مع اجراء اختبار المراقبة مرة اخرى والاعادة حتى يتم الوصول الى عنصر مماثل للمنتجه المدخل، وإذا لم يتم الوصول الى هذا العنصر فان الشبكة سوف تقرر ان هذا المنتجه المدخل غير معروف من قبل للشبكة ويتم ضمه وخزنه فى عنصر جديد من عناصر المخرج.

ومما سبق تتضح ديناميكية انتقال البيانات بين المدخل والمخرج، حيث يقوم المنتجه بالتردد بين الطبقتين عدة مرات محدثا ظاهرة تشابه ظاهرة الرنين فى الدوائر الالكترونية والتي تحمل فى طياتها بعض التعقيد ومع ذلك فان الخوارزميات تعتبر سهلة عند التنفيذ.

(٢٠-٤) خوارزم التعلم لنظرية الرنين المتكيف

(ART Network Learning Algorithm)

تشمل خوارزميات ليبمان (Lippmann) لشبكات نظرية الرنين المتكيف (رقم ١)

(ART1) على الخطوات الآتية:

١ - بدء التنشيط كمايلى : نفرض ان هناك شبكة شكل (٢٠-٦) وان:

أ - عدد العناصر فى طبقة المدخل (طبقة المقارنة) = (N)

ب - عدد العناصر الاخرى فى طبقة المخرج (طبقة التعرف) = (M)

ج- قيمة متجه الاوزان فى اتجاه التغذية الامامية (من المدخل الى المخرج) $w_{ij}(t)$ وذلك عند زمن (t) .

د- قيمة متجه الاوزان فى اتجاه التغذية العكسية (من المخرج الى المدخل) $t_{ij}(t)$ وذلك عند زمن (t) كما هو مبين فى شكل (٢٠-٦) حيث (i) هو احد العناصر فى طبقة المدخل و (j) هو احد عناصر طبقة المخرج وعلى ذلك فانه يمكن كتابة الآتى :

$$t_{ij}(0) = 1$$

$$w_{ij}(0) = \frac{1}{1+N}$$

$$0 \leq i \leq N-1 \quad 0 \leq j \leq M-1$$

ويجب تحديد قيمة معامل حد اختبار المقارنة (p) والذي يوضح مدى قرب التشابه بين

$$0 \leq p \leq 1$$

المتجه المدخل والمتجه المخزون كمايلي:

٢- يتم ادخال مداخل ذات قيم جديدة.

٣ - يتم حساب المواءمة كمايلي : وذلك بحساب متجه المخرج (μ_j) من العنصر (j) كمايلي :

$$\mu_j = \sum_{i=0}^{N-1} w_{ij}(t)x_i$$

مع الاخذ فى الاعتبار الحالة الآتية:

$$0 \leq j \leq M-1$$

وان (x_i) هى الجزء من المدخل (i) والذي يمكن ان ياخذ احد القيم اما القيمة صفر او القيمة واحد.

٤ - تحديد قيمة المتجه المناسب لافضل مواءمة، وذلك فى الصورة : $\mu_j^* = \max_j [\mu_j]$

٥ - إجراء الاختبار كمايلي:

$$\|X\| = \sum_{i=0}^{N-1} x_i \quad \text{أولا : حساب متجه التغذية الامامية فى الشكل :}$$

ثانيا : حساب حاصل الضرب النقطى بين متجه التغذية الامامية ومتجه التغذية

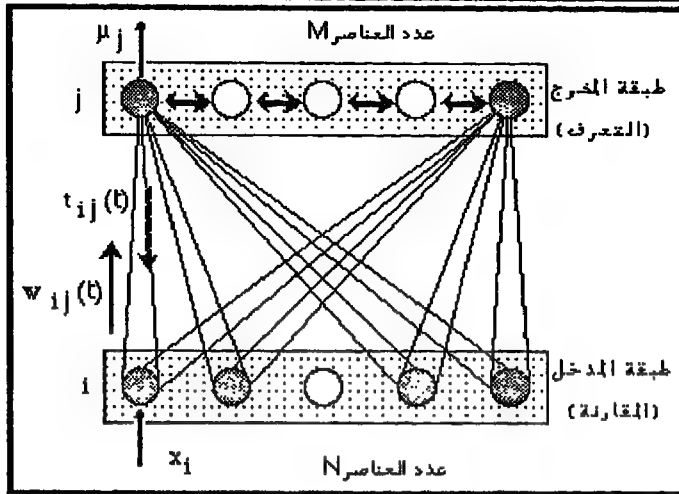
$$\|T \cdot X\| = \sum_{i=0}^{N-1} t_{ij}(t)x_i \quad \text{العكسية فى الشكل :}$$

ثالثا : اجراء المقارنة بين النسبة (S) والقيمة الحدية لمراقبة الاختبار (p) كمايلي :

$$is \frac{\|T \cdot X\|}{\|X\|} > P$$

YES go to 7

NO go to 6



شكل (٢٠-٦) متجهات التغذية الامامية والعكسية المثلة للربط

٦ - انهاء عملية افضل موائمة وذلك بوضع العنصر المستخدم قبل ذلك للموائمة يساوى الصفر، ثم الذهاب الى الخطوة رقم (٣).
go to 3.

٧ - اجراء الضبط لعملية افضل موائمة وذلك كمايلي :

$$t_{ij}^*(t+1) = t_{ij}^*(t) \times x_i$$

$$w_{ij}^*(t+1) = \frac{t_{ij}^*(t) \times x_i}{0.5 + \sum_{i=0}^{N-1} t_{ij}^*(t) \times x_i}$$

٨ - اجراء التنشيط لجميع العناصر الخاملة ثم الرجوع الى الخطوة الثانية لاختبار قيم مداخل جديدة.
go to 2.

(٢٠-٤-١) مثال لدورة التعليم (Example of A Learning Cycle)

نفرض ان هناك شبكة مكونة من عدد ثلاث عناصر فى طبقة المدخل وعدد عشرون عنصر

$$M = 20$$

$$N = 3$$

فى طبقة المخرج.

ويمكن شرح دورة التعليم فى الخطوات التالية:

١ - عند بدء التنشيط تكون الصورة كمايلي:

أ - اشارات التحكم (١) والتحكم (٢) تساوى الصفر.

ب- توضع جميع العناصر فى طبقة الخرج مساوية للصفر.

ج- تكون متجهات الاوزان فى حالتها الابتدائية ويجرى تحديد قيمة متجه الاتجاه الامامى

بالمعادلة الآتية :

$$w_{ij} = \frac{1}{1 + N}$$

حيث (N) هى رتبة المتجه المدخل (Input Vector).

٢ - يتم خزن كل الارقام الثنائية فى متجه التغذية العكسية للاوزان والذي يمثل بصمة ما فى شكل الرقم الثنائى الواحد.

٣ - يتم ادخال المتجه الجديد (x_1) عند المدخل ولكى تعمل الطبقة الاولى كطبقة ادخال، فان ذلك يلزم ان تكون اشارة التحكم تساوى الرقم الثنائى الواحد وتطبيق قاعدة الثلاثين نجد انه بتطبيق قانون (و) المنطقية على قيمة الاشارة (١) وقيمة المتجه عند المدخل (١) سوف تكون النتيجة (١) ايضاً وبذلك يتم السماح للمتجه المدخل بان يذهب الى الطبقة الثانية طبقة التعرف بدون اى تغيير.

٤ - فى هذه الطبقة يتم الموازنة بين المتجه القادم من طبقة المدخل ومتجه الاوزان للتغذية الازامية عند كل عنصر فى هذه الطبقة وذلك بايجاد حاصل الضرب النقطى بينهما وايجاد القيمة العظمى وتحديد العنصر الفائز.

٥ - يقوم العنصر الفائز فى طبقة التعرف بارسال المتجه المخزون به الى طبقة المدخل ثانية والتي سوف تعرف فى هذه الحالة بطبقة المقارنة مع تغير قيمة اشارة مصدر التحكم (١) الى الصفر.

٦ - ومن الواضح انه توجد ثلاث اشارات عند مدخل كل عنصر من عناصر هذه الطبقة هى:
المتجه المدخل والمتجه الممثل للبصمة واطار مصدر التحكم كما يلى:

متجه المدخل (1,1,0)

متجه التعرف (1,1,1)

اطار مصدر التحكم (1,1,0)

والتي تعطى خرجاً طبقاً لقاعدة الثلاثين (وذلك بتطبيق قاعدة (و) المنطقية) لتعطى قيمة متجه المقارنة كالاتى : (1,1,0)

٧ - يتم ارسال متجه المقارنة (1,1,0) مع متجه المدخل (1,1,0) الى دائرة مراقبة الاختبار حيث يتم استخراج (Similarity Value) نسبة التشابه بين المتجهين وفى هذا المثال تكون هذه النسبة ١:١ حيث انهما متشابهان واذا تم مقارنة هذه النسبة بقيمة حد الاختبار (ρ) والتي يمكن اختيارها تساوى (0.8) نجد انه قد تم توصيف المتجه المدخل بطريقة صحيحة.

٨ - بعد اجراء مراقبة حد الاختبار يتم تحديث متجه الاوزان فى العنصر الفائز وذلك ليحمل صفات المتجه المدخل والذي يجرى باستخدام وتطبيق قاعدة (و) (AND) المنطقية كما يلى:

$$T_{ij \text{ new}} = T_{ij \text{ old}} \text{ (AND) } x_i \\ = (1,1,1) \text{ (AND) } (1,1,0) = (1,1,0)$$

وبذلك نكون قد قمنا بخزن متجه المدخل (x_1) الممثل لبصمة جديدة في العنصر الفائز في طبقة التعرف.

٩ - وإذا فرضنا اننا سوف نقوم بتعليم الشبكة على متجة آخر جديد وليكن (x_2) ونريد ان

نعرف اين يمكن خزنه في طبقة التعرف في عنصر آخر فاننا سوف نقوم باتباع الخطوات التالية:

أ - نفرض المتجه الجديد ($1,0,1$) = (x_2) وعند اجراء المواءمة له نجد ان العنصر الفائز في

هذه الحالة ايضا هو العنصر السابق الذي يستخدم للتعرف على (x_1) ويتم امرار

المتجه الخاص بالعنصر الاول الى طبقة المقارنة لاجراء اختبار حد المقارنة وعند حساب

معامل التماثل (S) نجد ان قيمة اقل من قيمة المعامل (ρ) كمايلي:

$$s = \frac{\sum T_{ij} x_i}{\sum x_i} = 0.5$$

١٠ - وينشأ وضع متناقض وهو اختيار العنصر الفائز نفس العنصر الاول مع وجود تصنيف غير

صحيح وتدخل الشبكة مرحلة الخمود ويتم تحويل كل العناصر الي حالة الصفر ماعدا

العنصر الفائز.

١١ - تدخل الشبكة بعد ذلك دور التصنيف مرة اخرى الى ان يتم اختيار عنصر جديد ليكون

العنصر الفائز الثاني حيث يتم خزن ملامح المتجه المدخل الثاني.

ومن الواضح ان دخول متجه جديد لا يؤثر على المتجهات المخزونة قبل ذلك في طبقة

التعرف وهذه احدى الملامح الرئيسية للشبكات العصبية ذات الرنين والمتكيفة مع الوضع ذاتياً

علاوة على قصر زمن التعلم بالمقارنة بالانواع الاخرى من الشبكات.

تم بحمد الله

قائمة المراجع

- (1) Waterman D.A "A Guide to Expert Systems" , (Addison Wesley, Reading, Mass 1986)
- (2) Newell A. and Simon H. "Human Problem Solving", Englewood Cliffs, NJ., Prentice Hall (1972)
- (3) Davis R. "Expert Systems, Where are We ? and Where do we go from here ? , AI Memo No. 665, MIT AI Laboratory, (1982)"
- (4) Aikens J.S., Kunz J.C., and Shortliffe E.H., "PUFF: An Expert System for Interpretation of Pulmonary Function Data", Computers and Biomedical Research Vol. 16 (1983), PP 199-208.
- (5) A.Barr and E.A. Feigenbaum, "The Handbook of Artificial Intelligence", Vol. 1 Kaufmann, Los Altos, CA (1981).
- (6) S.R. Johnson, J.H. Connolly and E.A. Edmonds "Spectrogram Analysis: A Knowledge based Approach to Automatic Speech Recognition, In M.A. Bramer (ed), Research and Development in Expert Systems, Cambridge University Press, Cambridge, UK (1985).
- (7) P.Gilmore "A Proof Method for Quantification Theory", IBM J. Res. Devel. 4, 28-35 (1960).
- (8) Hanafusa 48 and H.Inoue (eds.), Robotics Research: The Second International Symposium. MIT Press, Cambridge, MA, MIT Press Series in Artificial Intelligence (1985).
- (9) IEEE Conference on Robotics and Automation, IEEE Computer Society, IEEE Computer Society Press, Silver Spring, MD (1985).
- (10) M.Brady, J.M. Hollerbach, T.L. Johnson, T. Lozano Perez, and M.T. Mason "Robot Motion: Planning and Control, MIT Press, Cambridge, MA" (1982).
- (11) P.P. Bonissone and R.M. Tong "Editorial: Reasoning with Uncertainty in Expert Systems", Int. J.Man-Match. Stud. 22(3), 241-250, March (1985).
- (12) L.A. Zadeh "A Computational Approach to Fuzzy Quantifiers in Natural Languages", Computer Mathematics, Appl.9(1), 149-184 (1983).
- (13) B.G. Buchanan and E.H. Shortliffe "Rule-based Expert Systems", Addison Wesley, Reading MA. (1984).
- (14) S.M. Ornstein, B.C. Smith and L.A. Suchman "Strategic Computing", Bull. Atom. Sci. 40, 11-15, December (1984).
- (15) H.A. Hunt and T.L. Hunt "Human Resource Implications of Robotics", Upjohn Institute for Employment Research, Kalamazoo, MI. (1983).
- (16) G.J. Agin and T.O. Binford "Computer Description of Curved Objects", Proceedings of the Third International Joint Conference on Artificial Intelligence, Palo Alto, CA, PP. 629-640 (1973).
- (17) P.H. Winston "Artificial Intelligence", Addison Wesley, Reading, MA. (1984).
- (18) E.Charniak and D. McDermott "Introduction to Artificial Intelligence", Addison Wesley, Reading, MA. (1985).
- (19) R. Beale and T. Jackson "Neural Computing: An Introduction", Department of Computer Science, University of New York (1990).
- (20) M. James BSP "Pattern Recognition", Professional Books, Oxford (1987).
- (21) T.Kohnen "Self Organization and Associative Memory", Third Eddi-

-
- tion, Springer verlage (1990).
- (22) T. Sejnowski, C.R. Rosenberg "Parallel Networks that Pronounce English Text", Complex Systems (1987).
- (23) S. Amari & M.A. Arbib "Competition and Cooperation in Neural Nets", Lecture Notes in Biomath, Vol. 45, Springer Verlag (1982).
- (24) J.J. Hopfield "Proc. Natl.", Acad. Sci., USA, Vol. 81. pp. 3088-3092 (1984).
- (25) Turbo Prolog, Porland.
- (25) M.A.El-Sharkawy, Published Articles in Computer Magazine, Dar El-Maarf, from No.16, (May 1987) up to No.39 (August 1990) :
- (د.محمد على الشرقاوي ، سلسلة مقالات ، مجلة كمبيوتر دار المعارف الاعداد من ١٦ الى ٣٩).
- 1- Expert Systems- The natural result of Artificial Intelligence developments.
(نظم الخبرة - احدى ثمرات تقدم الذكاء الاصطناعي)
 - 2- Computer Vision Expert System.
(نظم الخبرة للرؤية بالحاسب)
 - 3- CAD-CAM Expert Systems.
(نظم الخبرة للتصميم والانتاج)
 - 4- Forecasting and Decision Making Expert System.
(نظم الخبرة للتنبؤ واتخاذ القرار)
 - 5- Programming and Simulation Tools for Expert System Design).
(أساليب البرمجة والمحاكاة لتصميم النظم الخبرة)
 - 6- Tactical Air Target Recommender Expert Systems.
(نظم الخبرة التكتيكية لاختيار الاهداف للطيران)
 - 7- Some Objectives of Computer Teaching in Secondary Schools.
(بعض أهداف تعليم الحاسبات فى المدارس الثانوية)
 - 8- Expert System Development in Personal Computer Environment .
(تطور نظم الخبرة فى بيئة الحاسبات الشخصية)
 - 9- General Features of AI/Expert Systems Programming Languages :
a- List Programming (LISP).
b- Programming By Logic (PROLOG).
c- Object Oriented Programming).
(السمات العامة للغات البرمجة للذكاء الاصطناعي ونظم الخبرة)
أ - ليسب ب - برولوج ج - البرمجة الشيئية
 - 10- Artificial Intelligence (الذكاء الاصطناعي)
 - 11- Fields of Artificial Intelligence. (مجالات الذكاء الاصطناعي)
 - 12 - Knowledge Systems.
(نظم المعارف)
 - 13- Heuristic and Symbolic Processing.
(المعالجة الهرمية والرمزية)
 - 14- Building Expert Systems Tools.
(أدوات بناء النظم الخبرة)
 - (26) M.F.Tolba, "Computer and AI", Delta Computer Center, (1994).
(د.محمد فهمي طلبة الحاسب والذكاء الاصطناعي - مجموعة كتب دلتا)
 - (27) M.A.El-Sharkawy, "On the Implementation of Artificial Neural Networks in Image Processing", 3rd International Conference on Artificial Intelligence Applications, Cairo, Egypt, Jan. 3-7 (1995).
-

مركز الذكاء الاصطناعي للحاسبات Artificial Intelligence Computer Center

إن التقدم الكبير في مجال تكنولوجيا الذكاء الاصطناعي والشبكات العصبية الاصطناعية في القرن الحالي وبداية القرن القادم قد تطلب انشاء مراكز متخصصة في هذا المجال تولى اهتمامها لنقل وتطوير وتطويع هذه التكنولوجيات للعمل في البيئة العربية. ويعتبر مركز الذكاء الاصطناعي للحاسبات هو أول مركز متخصص في هذا المجال، وتشمل اتجاهات واهتمامات عمل المركز مايلي :

- ١ - تصميم البرامج التطبيقية المتعلقة بنظم الذكاء الاصطناعي وقواعد البيانات الديناميكية والنظم الخبيرة في المجالات العلمية والتعليمية والطبية والصناعية والتجارية وغيرها
 - ٢ - يولى المركز اهتماما خاصا بتطبيقات الشبكات العصبية الاصطناعية في مجالات التعرف والتنبؤات المالية للمشروعات الصناعية والتجارية المختلفة.
 - ٣ - يقوم المركز بتقديم الاستشارات الفنية المتخصصة واجراء الأبحاث العلمية المرتبطة بالذكاء الاصطناعي والشبكات العصبية الاصطناعية.
 - ٤ - عقد الدورات التدريبية والندوات العلمية في المجالات التطبيقية المرتبطة بهذا المجال في البيئة العربية.
 - ٥ - استخدام تقنيات الوسائط المتعددة والذكاء الاصطناعي في تصميم ونشر البرامج التعليمية، وتصميم وتنفيذ التجارب العملية.
- يقدم الكتاب هذا الفرع من علوم الحاسبات " الذكاء الاصطناعي والشبكات العصبية الاصطناعية " بشكل مبسط يفى بمتطلبات القارئ العادى والقارىء المتخصص، ويمكن أن يستخدم كمقرر في الذكاء الاصطناعي والشبكات العصبية في كليات الهندسة والعلوم والتجارة وأقسام الحاسبات ونظم المعلومات بالكليات المختلفة. يعتبر هذا الكتاب الأول ضمن سلسلة كتب يصدرها المركز في اتجاه تبسيط علوم وتكنولوجيا حاسبات المستقبل.
- وبالله التوفيق، ...

مركز الذكاء الاصطناعي للحاسبات

٨ شارع بن الوردى - مصر الجديدة - النزهة - القاهرة تليفون : ٢٤٥١٨٩٣
8 Ebn El-Wardy Street, Heliopolis, Nozha, Cairo. \ Tel.: 2451893

بريد الكتروني: MAMELSH@EGFRCUVX E.Mail:

الذكاء الاصطناعي والشبكات العصبية

Artificial Intelligence and Neural Networks

الجزء الثالث

لغات البرمجة والتطبيقات

Programming Languages and Applications

- الذكاء الاصطناعي ولغات البرمجة
- AI and Programming Languages
- تطبيقات باستخدام لغات المنطق (البرولوج السريع)
- Application using Turbo Prolog
- المعالجة الرمزية للعمليات الحسابية
- Symbolic Processing for Arithmetic Operations
- قواعد البيانات الديناميكية
- Dynamic Databases
- نظم تطبيقية
- Application Systems
- لغة البرمجة بأسلوب القائمة
- LISP

الجزء الرابع

الشبكات العصبية الاصطناعية والحساب العصبي

Artificial Neural Network & Neural Computing

- تطور الشبكات العصبية الاصطناعية
- Artificial Neural Networks Development
- الشبكات العصبية المتعددة الطبقات
- Multilayer Neural Networks
- شبكات كوهنن ذاتية التنظيم
- Kohonen Self-Organising Neural Networks
- شبكات هوبفيلد ذات الاتصال الكامل
- Hopfield Fully-Connected Neural Networks
- الشبكات العصبية الكثيفة التوازي
- Massively Parallel Neural Networks

الجزء الاول

الذكاء الاصطناعي الحاضر والمستقبل

Artificial Intelligence the Present & the Future

- الذكاء الاصطناعي والذكاء الانساني
- Artificial and Human Intelligence
- مجالات الذكاء الاصطناعي
- Domains of Artificial Intelligence
- النظم الخبيرة ومجالاتها المختلفة
- Expert Systems and its Different Domains
- المجالات التطبيقية للنظم الخبيرة
- Expert System's Practical Applications
- حاسبات الجيل الخامس
- Fifth Generation Computer Systems

الجزء الثاني

المعالجة الرمزية

Symbolic Processing

- الأسس الرياضية للمعالجة الرمزية
- Mathematical Basics For Symbolic Processing
- النمذجة الحسابية وتهيئة المعارف
- Computational Modelling and Knowledge Representation.
- آليات تقنيات الاستدلال
- Toolbox of Inference Techniques
- نظم الاستدلال المعتمدة على القواعد
- Rule-Based Inference Systems